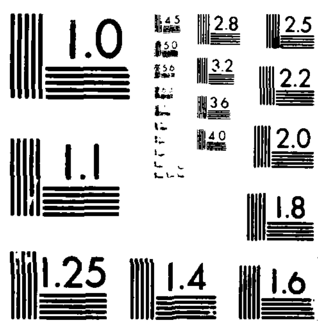
AD-A085 436   TEXAS UNIV AT EL PASO DEPT OF ELECTRICAL ENGINEERING   F/G 7/4
COMPUTERIZED METHOD FOR THE GENERATION OF MOLECULAR TRANSMITTAN--ETC(U)
APR 80 J H PIERLUISSI, K TOMIYAMA   DAAG29-79-C-0067

UNCLASSIFIED   ERADCOM/ASL-CR-80-0067-1   NL

1 OF 2
AD
A085436



MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS-1963-A

ASL-CR-80-0067-1

LEVEL II

AD A

Reports Control Symbol
OSD-1366

COMPUTERIZED METHOD FOR THE GENERATION OF MOLECULAR TRANSMITTANCE FUNCTIONS IN THE INFRARED REGION

APRIL 1980

Prepared by

Joseph H. Pierluissi
Ken Tomiyama

Electrical Engineering Department
The University of Texas at El Paso
El Paso, Texas 79968

Under Contract: DAAG29-79-C-0067

Contract Monitor: JAMES T. HALL

US Army Electronics Research and Development Command
ATMOSPHERIC SCIENCES LABORATORY
White Sands Missile Range, NM 88002

NOTICES

Disclaimers

The findings in this report are not to be construed as an official Department of the Army position, unless so designated by other authorized documents.

The citation of trade names and names of manufacturers in this report is not to be construed as official Government indorsement or approval of commercial products or services referenced herein.

Disposition

Destroy this report when it is no longer needed. Do not return it to the originator.

DISCLAIMER NOTICE

THIS DOCUMENT IS BEST QUALITY PRACTICABLE. THE COPY FURNISHED TO DTIC CONTAINED A SIGNIFICANT NUMBER OF PAGES WHICH DO NOT REPRODUCE LEGIBLY.

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER ASL-CR-80-0067-1	2. GOVT ACCESSION NO. AD-A085436	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) 6 Computerized Method for the Generation of Molecular Transmittance Functions in the Infrared		5. TYPE OF REPORT & PERIOD COVERED 9 Final Report.
		6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s) 10 Joseph H. Pierluissi Ken Tomiyama		8. CONTRACT OR GRANT NUMBER(s) 15 DAAG29-79-C-0067
9. PERFORMING ORGANIZATION NAME AND ADDRESS Electrical Engineering Department The University of Texas at El Paso El Paso, Texas 79968		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS
11. CONTROLLING OFFICE NAME AND ADDRESS Department of the Army U.S. Army Research Office		12. REPORT DATE 11 Apr[redacted]80 13. NUMBER OF PAGES 180
14. MONITORING AGENCY NAME & ADDRESS(if different from Controlling Office) Atmospheric Sciences Laboratory White Sands Missile Range, NM 88002		15. SECURITY CLASS. (of this report) Unclassified
		15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release: Distribution unlimited

12 195

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18 ERADCOM/ASL 19 CR-80-0067-1

18. SUPPLEMENTARY NOTES

Contracting Officer Technical Representative: Dr. Leo Alpert, P.O. Box 12211, Research Triangle Park, N.C. 27709

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Molecular Absorption, Trace Gases, Modularized Lowtran

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

A study is made of two basically distinct methods normally used in the development of band models for the calculation of gaseous molecular transmittance in the infrared region. The first method consists of the determination of the "empirical" transmittance function and the associated absorber and spectral parameters from measured or calculated transmittance spectra. The second method consists of the determination of the absorber and spectral parameters

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE

408579

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered) with an assumed "analytical" transmittance function, using the same type of data. Computerized numerical techniques are presented in connection with the first method and a generalized transmittance function is adopted for the second method. Although the methodology is generally applicable to other gaseous species, it is specifically discussed in connection with the trace gases SO_2, NO, NO_2, and NH_3. As a secondary effort a structural breakdown of the Lowtran code is presented for the purpose of incorporating the band models for the trace gases. The code is separated into basic functional modules or subroutines controlled by a main program. The modularization itself was primarily performed under a separate effort through the Atmospheric Sciences Laboratory.

TABLE OF CONTENTS

Table of Contents (continued)

Page

Table of Contents (continued)

Page

LIST OF TABLES

List of Tables (continued)

LIST OF FIGURES

I. Introduction

Following the efforts of Elssaser[1] numerous workers have attempted to arrive at computationally-simple models for gaseous molecular transmittance, averaged over narrow spectral intervals in the infrared. These efforts may be naturally divided into those involving the analytical derivation of a mean transmittance function from Beer's Law, and those involving the extraction of the transmittance function itself from transmittance data. Traditionally, the former are called "analytical" and the latter are called "empirical". The method normally used in the empirical models consists of the extraction of the transmittance function through graphical techniques, with the adoption of a relationship between spectral and absorber parameters. In the development of analytical models a transmittance function is adopted at the offset, and the spectral and absorber parameters are afterward determined through computerized numerical procedures.

In the work reported here the authors present a totally computerized version of the classical graphical methods for the extraction of the empirical transmittance function. This is followed by a presentation of a numerical method which uses a double-exponential transmittance function for the development of analytical band models. Both methods are then applied to 20 cm^{-1} averaged line-by-line transmittance data for the atmospheric trace gases SO_2, NO, NO_2, and NH_3. The model parameters are listed at 5 cm^{-1} intervals throughout the major absorption bands of these gases for the convenience of the community of band model users. Although the methodology is applied specifically to the trace gases, no restrictions are immediately evident in the extension to other gaseous absorbers in the infrared. In fact, the analytical method was successfully applied earlier[2] to the principal band centers of the major absorbers H_2O vapor, O_3 and the uniformly-mixed gases.

As an application of the results found through this effort, the band models for the trace gases were incorporated in the widely-used code called Lowtran. To facilitate the inclusion of these models, as well as of others, the code was broken down into separate subroutines or modules controlled by a master program. The subroutines include the evaluation of the equivalent absorber amount, the selection of the spectrally-effective attenuation model and the individual attenuation models. The principal purpose of the modularization is to assist users with the modification of the code to suit their individual requirements on transmission models.

II. The Transmittance Equation

The monochromatic transmittance τ_ν at frequency ν for the passage of infrared radiation through a path length Z in an inhomogeneous medium with pressure and temperature distributions P(Z) and T(Z), respectively, is given by Beer's Law in the form

$$\tau_\nu = e^{-\int K_\nu(P,T)dU(Z)} \tag{1}$$

where K_ν is the resultant absorption coefficient for all contributing lines and gaseous absorbers, and U is the absorber amount. For broadband radiation detected by an instrument of spectral response ϕ_ν, the variable of interest is the weighted mean transmittance τ, defined as

$$\tau = \int \tau_\nu \phi_\nu d\nu \Big/ \int \phi_\nu d\nu \tag{2}$$

Equation (2) has been evaluated analytically over a spectral interval $\Delta\nu$ for the special case of Lorentzian broadened lines having assumed line distributions and intensities, leading to the classical band models[1,3]. Numerous variations of the classical band models may be found in the literature, most of which specify the analytical form of τ in terms of mean line or meteorological variables. A notable exception is the model of King[4] which expresses the homogeneous-path transmittance as

$$\tau = g(S\alpha^n U), \tag{3}$$

where g is a function to be determined empirically, S is the mean line intensity, α is the mean line half-width and n is an absorber parameter with the physical constraints of zero and one in the weak-line and strong-line limits, respectively. The path inhomogeneity may be accounted for in Eq. (3) through the Curtis-Godson equivalences

$$S\alpha^n U = \int S(Z)\alpha^n(Z)dU(Z). \qquad (4)$$

From practical considerations, it is often desirable to transform the argument in Eq.(3) with the known relations

$$S = S_o \left(\frac{T_o}{T}\right)^a \qquad (5)$$

$$\alpha = \alpha_o \left(\frac{P}{P_o}\right)\left(\frac{T_o}{T}\right)^{\frac{1}{2}} \qquad (6)$$

in order to obtain

$$\tau = g\left\{ C\left(\frac{P}{P_o}\right)^n \left(\frac{T_o}{T}\right)^m U \right\}, \qquad (7)$$

where C is a spectral parameter combining S_o and α_o^n, m is an absorber parameter combining the temperature exponents of S and α, a is an absorber constant, and the subscript "o" denotes standard conditions. For computational convenience Eq. (7) may be expressed as

$$\tau = f\{x\}, \qquad (8)$$

where

$$x = C' + \log_{10} W \tag{9}$$

$$C' = \log_{10} C \tag{10}$$

$$W = \left(\frac{P}{P_o}\right)^n \left(\frac{T_o}{T}\right)^m U \quad . \tag{11}$$

Here, f is the transmittance function, C' is the spectral parameter, W is the equivalent absorber amount, and n and m are the absorber parameters; all of which are to be determined from transmittance data for each absorber.

III. Computerized Method of Empirical Model Development

3.1 Introduction

Assuming the availability of equal transmittance data, which is defined below, we have developed an algorithm, called ADSET, which evaluates absorber parameters n, m, spectral parameters $C'(\nu)$ and an empirical transmission function simultaneously. In the algorithm the transmission function is linearized and a linear regression technique is utilized for parameter evaluation. In order to evaluate the band model parameters and the empirical transmission function simultaneously, a set of auxiliary variables are introduced. Each data point is identified through the auxiliary variables to an absorption band and to a transmittance 'cut'. This enables us to obtain globally optimal set of parameters and the empirical transmission function simultaneously.

Based on the derived optimal pointwise transmission function, a piecewise analytical transmission function is developed. The commonly used computer code Lowtran for the evaluation of atmospheric transmittance can be greatly simplified by the use of this piecewise analytical transmission function to model the major absorbers.

Finally, the code ADSET also contains a subroutine which can compute the spectral parameter value $C'(\nu)$ for non-major absorption bands.

3.2 Data Structure

Several transmittance values τ_j, j=1, 2, ..., NCUT are chosen a priori, where NCUT is the number of chosen transmittance values. Curves of growth data (i.e. τ versus U) for each layer of atmosphere are assumed to be given at these transmittance values. Therefore, the curves of growth have 'cut' structure, namely, all data points are on one of the transmittance cuts $\tau = \tau_j$, j=1, 2, ..., NCUT (See Fig. 1). We call a data set with this cut structure an 'equal transmittance' data in the sequel.

3.3 Linearization of Transmission Function

Since f in Eq.(8) is known to be strictly monotone decreasing from one to zero as x changes from $-\infty$ to ∞, there exists an inverse function f^{-1} defined on (0,1) such that

$$\begin{aligned} x &= f^{-1}(\tau) \\ &= C' + \log W \\ &= C' + n\log(\frac{P}{P_o}) + m\log(\frac{T_o}{T}) + \log U. \end{aligned} \tag{12}$$

Let us define x_j, j=1, 2, ..., NCUT be the inverse image of the prechosen transmittance values τ_j, j=1, 2, ..., NCUT i.e.,

$$x_j = f^{-1}(\tau_j), \quad j= 1, 2, \ldots, \text{NCUT}, \tag{13}$$

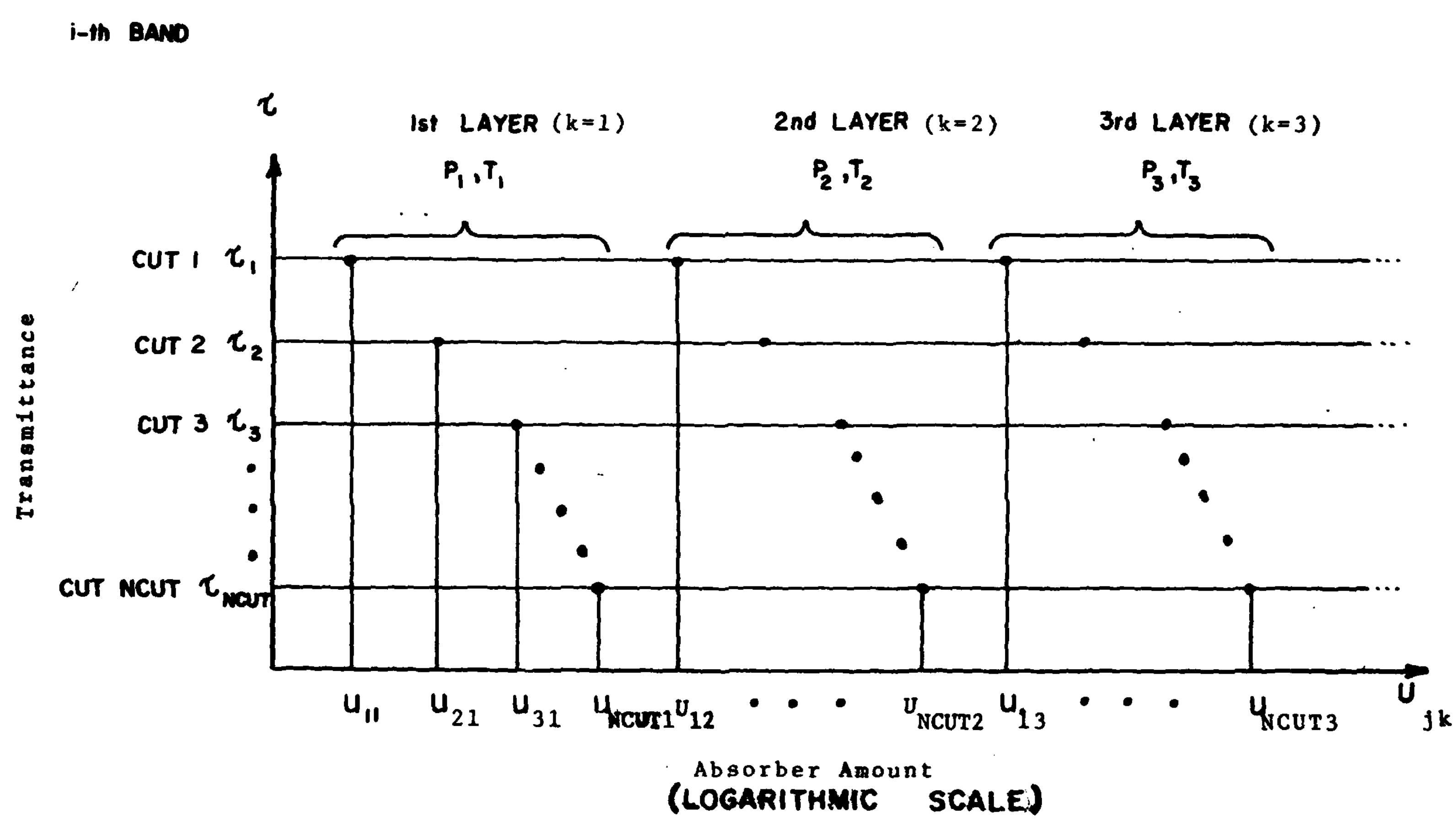


Fig. 1. Schematic representation of "equal transmittance" data structure.

Then, the set of points (x_j, τ_j), $j = 1, 2, \ldots,$ NCUT is nothing but the empirical transmission function, which is to be found. From Eq. (12), we reach the following regression equation.

$$n \log(\frac{P}{P_o}) + m \log(\frac{T_o}{T}) + C' - x = -\log U. \qquad (14)$$

Note that this equation is linear in the unknown parameters n, m, C' and x. Therefore, the linear regression technique can be used to evaluate the optimum values for the parameters.

3.4 Formation of the Square Error

The square error corresponding to the k-th data point in i-th absorption band's j-th cut, denoted by E_{ijk}, is given by

$$E_{ijk} = \{n\log(\frac{P_{ijk}}{P_o}) + m\log(\frac{T_o}{T_{ijk}}) + C_i' - x_j - (-\log U_{ijk})\}^2 \qquad (15)$$

Hence, the total square error E_{ij} for this cut is

$$E_{ij} = \sum_{k=1}^{L_{ij}} E_{ijk}, \qquad (16)$$

where L_{ij} is the number of layers in this cut. Similarly, the total square error E_i for i-th band and the grand total square error E are given by

$$E_i = \sum_{j=1}^{J_i} E_{ij} = \sum_{j=1}^{J_i} \sum_{k=1}^{L_{ij}} E_{ijk}, \qquad (17)$$

$$E = \sum_{i=1}^{NB} E_i = \sum_{i=1}^{NB} \sum_{j=1}^{J_i} \sum_{k=1}^{L_{ij}} E_{ijk}, \qquad (18)$$

where J_i and NB are the numbers of the cuts in i-th absorption band and of the absorption bands, respectively. The final expression can be simplified if we assume that the number of layers (L_{ij}) in every cut is equal to a constant L_i. For this case

$$E = \sum_{i=1}^{NB} \sum_{j=1}^{J_i} \sum_{k=1}^{L_i} E_{ijk}. \qquad (19)$$

Our objective is to find optimum set of parameters (n^*, m^*, $C'_1{}^*$, $C'_2{}^*$, ..., $C'_{NB}{}^*$, $x_1{}^*$, $x_2{}^*$, ..., $x_{NCUT}{}^*$) which minimizes this grand total error E.

3.5 Auxiliary Variables

In order to perform the minimization of E with respect to the above parameters simultaneously, we modify the square error E_{ijk} so that it contains all the parameters. This is done by introducing two sets of auxiliary variables u_i, i=1, 2, ..., NB and v_j, j=1, 2, ..., NCUT. Using them, E_{ijk} is redefined as

$$E_{ijk} = \{n\log(\frac{P_{ijk}}{P_o}) + m\log(\frac{T_o}{T_{ijk}}) + u_{1,ijk}\, C'_1 + \ldots + u_{NB,ijk} C'_{NB} + v_{1,ijk} K_1 + \ldots + v_{NCUT,ijk}\, K_{NCUT} - (-\log U_{ijk})\}^2 \qquad (20)$$

where $K_j = -x_j$, $j=1, 2, \ldots,$ NCUT. The auxiliary variables act as identifiers of the band and the cut. If a data point is for $\hat{i}$-th band's $\hat{j}$-th cut, then $u_{\hat{i}} = 1$ and $u_i = 0$ for all $i \neq \hat{i}$ and $v_{\hat{j}} = 1$ and $v_j = 0$ for all $j \neq \hat{j}$. Thus, only the spectral parameter $C'_{\hat{j}}$ and the cut parameter $K_{\hat{j}}$ corresponding to the current data are active and all other spectral and cut parameters disappear. Hence, Eq. (20) reduces to Eq. (15). The change from x_j to $K_j = -x_j$ is made in order to symmetrize the coefficient matrix of the resulting normal equation. This change makes it possible to utilize any specialized solution method for the symmetric normal equation when the space conservation is important.

3.6 Regression Analysis

Using the grand total error E with the redefined E_{ijk} in Eq. (20), the best parameter values n^*, m^*, C'^*_1, $\ldots$, C'^*_{NB}, $K^*_1, \ldots,$ K^*_{NCUT} are simultaneously determined by the linear regression. Setting the partial derivatives of E with respect to parameters equal to zero results in a linear normal equation of the form $AX = B$, where A, B and X are, respectively, a symmetric coefficient matrix, a constant vector and a parameter vector defined by

$$A = \left[\begin{array}{c|c|c}
\begin{matrix} \Sigma v^2_{NCUT} & & O \\ & \ddots & \\ O & & \Sigma v^2_1 \end{matrix} & * & * \\
\hline
* & \begin{matrix} \Sigma u^2_{NB} & & O \\ & \ddots & \\ O & & \Sigma u^2_2 \end{matrix} & * \\
\hline
* & * & \begin{matrix} \Sigma(\log \frac{T_o}{T})^2 & * \\ * & \Sigma(\log \frac{P}{P_o})^2 \end{matrix}
\end{array}\right] \tag{21}$$

$$B = \left[\Sigma(-v_{NCUT}\log U),\ \ldots,\ \Sigma(-v_1 \log U),\ \Sigma(-u_{NB}\log U),\ \ldots,\right.$$

$$\left.\Sigma(-u_2 \log U),\ \Sigma(-\log(\frac{T_o}{T})\log U),\ \Sigma(-\log(\frac{P}{P_o})\log U)\right]^t, \quad (22)$$

$$X = \left[K_{NCUT},\ \ldots,\ K_1,\ C'_{NB},\ \ldots,\ C'_2,\ m, n\right]^t \quad (23)$$

The * in Eq. (21) represents some nonzero elements. Also the Σ in the above equations represents the triple sum $\sum_{i=1}^{NB}\sum_{j=1}^{J_i}\sum_{k=1}^{L_i}$ in Eq. (19). One may realize that C'_1 does not appear in Eq. (23) and hence the corresponding auxiliary variable u_1 is also absent from Eqs. (21) and (22). This is because one of C'_1, ..., C'_{NB} is dependent on other C'_i so that C'_1, ..., C'_{NB} cannot be determined uniquely. It is necessary that one of C'_is be given a number *a priori*. Here C'_1 is chosen and is given the value zero, and therefore is eliminated from the parameter vector X. This choice calls for some explanation. On τ vs. logW diagram the optimum empirical transmission function can be placed anywhere. What it amounts to is that a different placement results in a different set of C'_i values which is a linear shift (addition or subtraction of a constant) of another set of C'_i values. Only the relative relationship among C'_i is unique. This is clearly indicated in Fig. 2.

Since the placement of the empirical transmission function is arbitrary, we may position it on the data points corresponding to the first absorption band. In other words, the first absorption band is taken as the

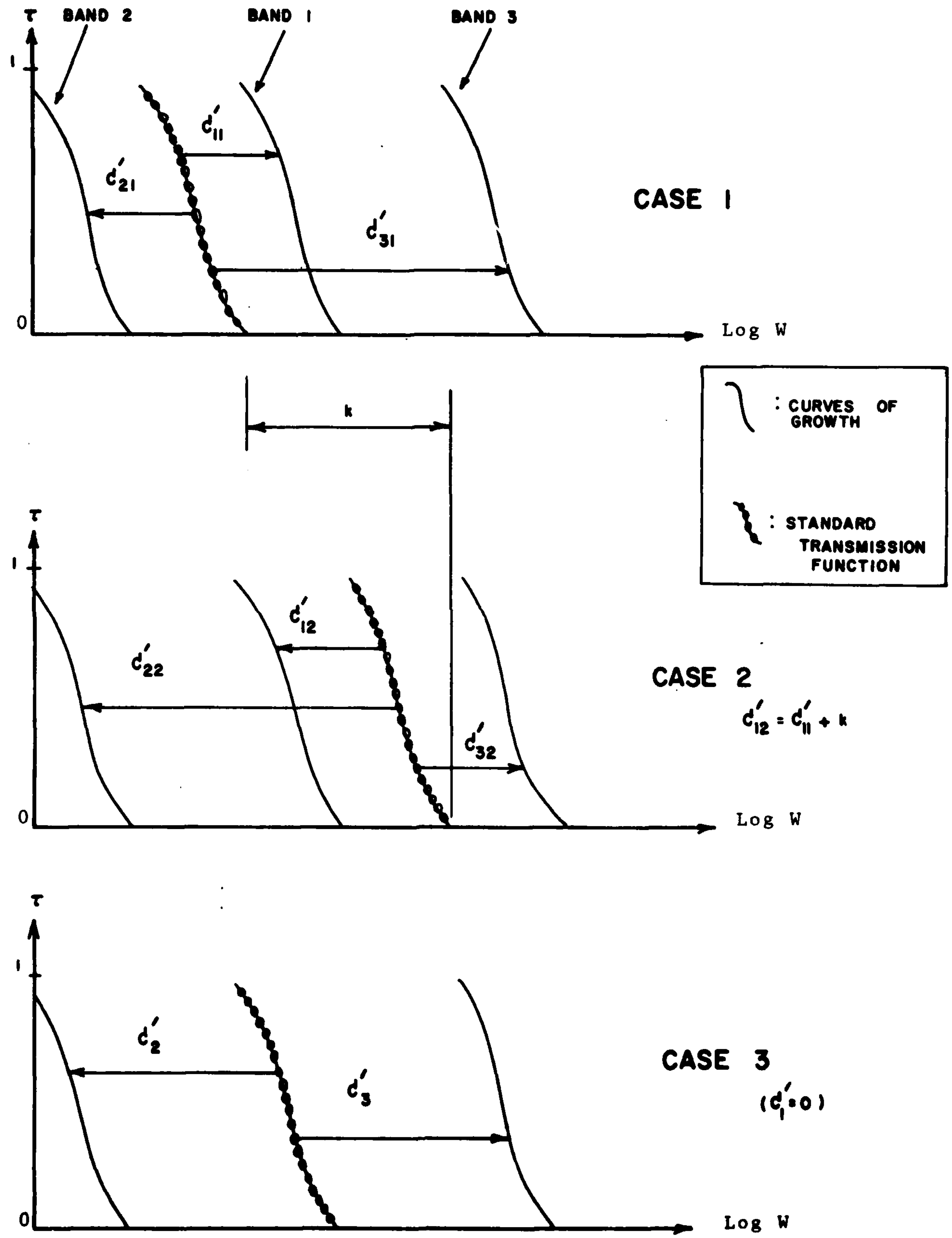


Fig. 2. Schematic representation of linear shift accounting for spectral dependence of transmittance.

reference band and the corresponding spectral parameter C_1' is set to be zero.

The queuing of parameters in X vector is determined in such a way that as many upper principal minor matrices as possible become diagonal (See Eq. (21)). This arrangement can reduce the amount of computation in the early stage of Gauss elimination steps when the normal equation is solved, and can result in less computational error.

3.7 Piecewise Analytical Transmission Function

After the best parameter values are computed, the piecewise analytical transmission function is generated by the piecewise interpolation. The transmittance region (0,1) is divided into NCUT - 1 subregions by the transmittance cuts $\tau_2, \tau_3, \ldots, \tau_{NCUT-1}$. Let $\tau_1 > \tau_2 > \ldots > \tau_{NCUT}$, then the subregions are given by

Subregion 1	$[\tau_2, 1)$,
Subregion 2	$[\tau_3, \tau_1]$,
.	.
.	.
.	.
Subregion NCUT-2	$[\tau_{NCUT-1}, \tau_{NCUT-2}]$,
Subregion NCUT-1	$(0, \tau_{NCUT-1}]$.

The top and bottom subregions contain τ_1 and τ_{NCUT} as an inner point, respectively. The interpolation in each subregion is done by the double exponential function defined by

$$\tau(x) = \exp \{-10^{a_1 + a_2 x + a_3 x^2}\} . \qquad (24)$$

The generally-used linear interpolation is not used here since subregions cannot be assumed small enough for the linear approximation to be valid. Furthermore, the linear interpolation is totally inadequate for the top and bottom subregions. On the other hand, the double exponential function takes the values between and is asymptotic to one and zero as the argument varies from $-\infty$ to ∞. It is also known that this function closely approximates the standard empirical transmission function used in the Lowtran code.[2,16]

The parameters a_1, a_2, and a_3 for each subregion are determined by two different methods. The first method assumes that $a_3 = 0$ and uses no further data to compute a_1 and a_2. They are simply determined by the condition that the interpolation function in each subregion passes through the end points. In the top and bottom subregions, the function is required to pass through two points; (τ_1, x_1) and (τ_2, x_2) for the top and $(\tau_{NCUT-1}, x_{NCUT-1})$ and (τ_{NCUT}, x_{NCUT}) for the bottom subregions.

The second method does not assume that $a_3 = 0$ and requires additional data to compute parameter values. The same condition that each interpolation function passes through two points reduces the number of unknown parameters to one. The last parameter is determined by minimizing the subregional square error E_i defined by

$$E_i = \sum_{i=1}^{L_i} (\tau_i - \exp\{-10^{a_1 = a_2 x_i + a_3 x_i^2}\})^2 \tag{25}$$

for those data points in respective subregions.

3.8 C' for Non-major Bands

Finally, the spectral parameters C' for non-major bands are computed by a straightforward method. The discrepancies between x^*_i and $\log W_i$ values computed for all cuts for one band are averaged to obtain the spectral parameter $C'(\nu)$ for that band, i.e.,

$$C' = \frac{1}{N} \sum_{i=1}^{N} (x_i^* - \log W_i), \tag{26}$$

where W_i are computed by Eq. (11) with optimal n* and m*.

IV. Computerized Method of Analytical Model Development

4.1 Introduction

In the last chapter, we assumed no analytical form for the transmission function $\tau = f(x)$ when the standard transmission function was computed. Here, by assuming the double exponential form given by Eq. (24) as the transmission function for the entire transmittance range, we derive an algorithm which can evaluate the best function parameter values a_1, a_2, and a_3; together with the band model parameters n, m, and C_i'. Note that the double exponential function was used for the piecewise interpolation in the last chapter. But the computation of the function parameters was performed after the band model parameters and the empirical transmission function were obtained. In other words, the computation in the last chapter was sequential but not simultaneous. The algorithm we present in this chapter is, on the contrary, the simultaneous evaluation of all parameters. The preliminary development of this algorithm can be found in Ref. 5.

4.2 Basic Equations

The basic equations are Eq. (8) and Eq. (24) of the last chapter, which are cited here for the ease of reference.

$$\tau = f(x) \qquad (27)$$

$$f(x) = \exp\{-10^{a_1 + a_2 x + a_3 x^2}\}. \qquad (28)$$

Now, since we have assumed the function form, we can compute the transmittance if we have the value of x. Hence, we do not have to take the inverse function as we did before to perform the regression analysis. Instead, we take the square difference of the given and computed τ directly from this expression. Thus, we get

$$E_{ij} = [\tau_{ij} - \exp\{-10^{a_1+a_2x_{ij}+a_3x_{ij}^2}\}]^2 , \tag{29}$$

for i-th absorption band's j-th data point, where, as before, x_{ij} is given by

$$x_{ij} = C'_i + n\ \log(\frac{P_{ij}}{P_o}) + m\ \log(\frac{T_o}{T_{ij}}) + \log U_{ij}. \tag{30}$$

By summing this individual error for all data in i-th band, we have the total error for this band as

$$E_i = \sum_{j=1}^{J_i} E_{ij}, \tag{31}$$

where J_i is the number of data in i-th band.

Again, we introduce auxiliary variables u_i, i=1,2, ...,NB in order to introduce all C'_i, i=1,2,...,NB into the x_{ij} expression Eq.(30). By this we get

$$x_{ij} = \sum_{k=1}^{NB} u_{k,ij}C'_k + n\ \log(\frac{P_{ij}}{P_o}) + m\ \log(\frac{T_o}{T_{ij}}) + \log U_{ij}, \tag{32}$$

We use this expression for x_{ij} in the following total error E

$$E = \sum_{i=1}^{NB} E_i = \sum_{i=1}^{NB} \sum_{j=1}^{J_i} E_{ij}. \tag{33}$$

Now, we are ready to take partial derivatives with respect to the parameters n, m, C'_1, ..., C'_{NB}, a_1, a_2, and a_3 to form the normal equation for this regression problem. Theoretically speaking, we can evaluate the 'best' parameter values by solving the normal equation. But obviously the grand total Eq. (33), which is to be minimized, is not a quadratic function of the unknown parameters and, therefore, the resulting normal equation is not a linear function of them. Hence, we need to adopt a different numerical method for the evaluation of the 'optimal' parameter values.

4.3 Nonlinear Optimization Method

The computational technique we use here is a recursive technique which is referred to as the conjugate gradient method[17]. In essence, this technique improves a set of guesses of the parameter values recursively by locating a new set of guesses which yields smaller error. For a given guess $(\alpha^n, \beta^n, \ldots, \gamma^n)$ of the minimizing parameter vector, at which the error is minimized, the best direction of the search in the parameter space for a new guess is first determined using up to second order derivatives of the error. Then the one-dimensional search for the minimizing point is performed along this direction from $(\alpha^n, \beta^n, \ldots, \gamma^n)$ to find a new guess $(\alpha^{n+1}, \beta^{n+1}, \ldots, \gamma^{n+1})$

which yields locally the smallest error. Now this procedure is repeated recursively to obtain a sequence of guesses until the gradients become less than a small positive number which is chosen *a priori*.

Actual computation was done by utilizing the packaged subroutine FMCG in SSP library available from IBM[18]. The necessary gradients are

$$
\begin{aligned}
\frac{\partial J}{\partial a_1} &= -2\Sigma D_j \delta f_j, \\
\frac{\partial J}{\partial a_2} &= -2\Sigma D_j \delta f_j x_j, \\
\frac{\partial J}{\partial a_3} &= -2\Sigma D_j \delta f_j x_j^{\,2}, \\
\frac{\partial J}{\partial n} &= -2\Sigma D_j \delta f_j (a_2 + 2a_3 x_j)\log(\frac{P_j}{P_o}), \\
\frac{\partial J}{\partial m} &= -2\Sigma D_j \delta f_j (a_2 + 2a_3 x_j)\log(\frac{T_o}{T_j}), \\
\frac{\partial J}{\partial C'_i} &= -2\Sigma D_j \delta f_j (a_2 + 2a_3 x_j) u_i,
\end{aligned}
\tag{34}
$$

where, Σ represents $\sum_{i=1}^{NB} \sum_{j=1}^{J_i}$ and D_j and δf_j are given by

$$D_j = \{E_{ij}\}^{\frac{1}{2}}, \tag{35}$$

$$\delta f_j = (\ln 10)\, 10^{a_1 + a_2 x_j + a_3 x_j^2} f(x_j), \tag{36}$$

and f(x) is given by Eq. (28).

Note that there exists a linear dependence among the gradients which is

$$a_2 \frac{\partial J}{\partial a_1} + 2a_3 \frac{\partial J}{\partial a_2} = \sum_{i=1}^{NB} \frac{\partial J}{\partial C_i'} \tag{37}$$

Therefore, the parameter set $\{n, m, C_1', \ldots, C_{NB}', a_1, a_2, a_3\}$ cannot be determined uniquely. As it was explained in the previous section, this is due to the arbitrariness in the positioning of the standard transmission function. Hence, the spectral parameter C_1' is again set to be zero, so that we can evaluate unique set of optimal parameters.

V. Comparison of the Two Methods

5.1 Introduction

Both methods can evaluate the optimal n, m and $C'(\nu)$ values for major and non-major bands and also a standard transmission function. But there are some basic differences which are discussed in the sequel.

5.2 Final Products

The final product of the ADSET code is a piecewise analytical standard transmission function together with the band model parameters. Each analytical piece of the standard transmission function covers only one of the pre-chosen subintervals of (0,1) transmittance range. On the other hand, SIMMIN produces only one analytical transmission curve for the entire range. Therefore, ADSET has more flexibility to adjust to the transmittance curve variations. This feature of ADSET can be very valuable for the gases with non-standard curves of growth.

This difference is amplified when the number of the transmittance sub-regions used in ADSET is increased. However, as the number of subregions increase, the requirement on the usable data becomes severer and more spaces are necessary to store the computed results. Hence, the determination of the number of subregions should be resorted to compromise.

5.3 Installation of the Results in Lowtran

The final products of two codes ADSET and SIMMIN were installed in the widely-used Lowtran code, as discussed in Section VII of this report. The SIMMIN results require less memory space, less time for transmittance computation and simpler coding than the ADSET result. In fact, for the SIMMIN result, all that have to be stored are the five band model parameters n, m, a_1, a_2, and a_3, and a set of spectral parameters $C'(\nu_i)$ for each absorber. Furthermore, the computation of τ can be done by only one FORTRAN statement. On the other hand, the ADSET result requires the storage of NCUT-1 of a_1, a_2, and a_3 values, n and m and a set of $C'(\nu)$ values for each absorber. There can be a large difference in the number of the sets of a_1, a_2, and a_3 values to be stored. Moreover, some judging statements are necessary to select the right set of a_1, a_2, and a_3 for each transmittance computation.

5.4 Data Requirements

The ADSET code requires the cut structured data such that the transmittance of each data point must fall in one of prechosen values. But the SIMMIN code does not impose any conditions on the data set.

Some considerations on the requirement of equal transmittance data for ADSET are due here. Even if the available data do not have equal transmittance structure, it can be transformed into the required form using interpolation/extrapolation. This constitutes the pre-processing of the raw data. Curves of growth data τ vs. log U with τ values not necessarily coinciding with the prechosen values can be locally interpolated/extrapolated using an analytical function. This procedure is indicated in Fig. 3. Again, the double exponential function is an excellent choice for the interpolation function. We note that an almost exact technique as the one used in obtaining a piecewise analytical transmission function can be used for this purpose. In fact, only a minor modification of the interpolation subroutine used in ADSET can accomplish this task.

5.5 Computation Time

The numerical methods used in ADSET and SIMMIN for solving normal equations are essentially different. The method in SIMMIN is a recursive algorithm and the other in ADSET is a non-iterative one. Therefore, the computation time for ADSET is determined by the size of the data set only, whereas, the one for SIMMIN depends on both the actual data values and the initial guesses. It is difficult to estimate the computation time for SIMMIN due to this dependence. One way of controlling the time is to limit the number of iterations performed. This feature is included in the packaged subroutine FMCG which is used for actual

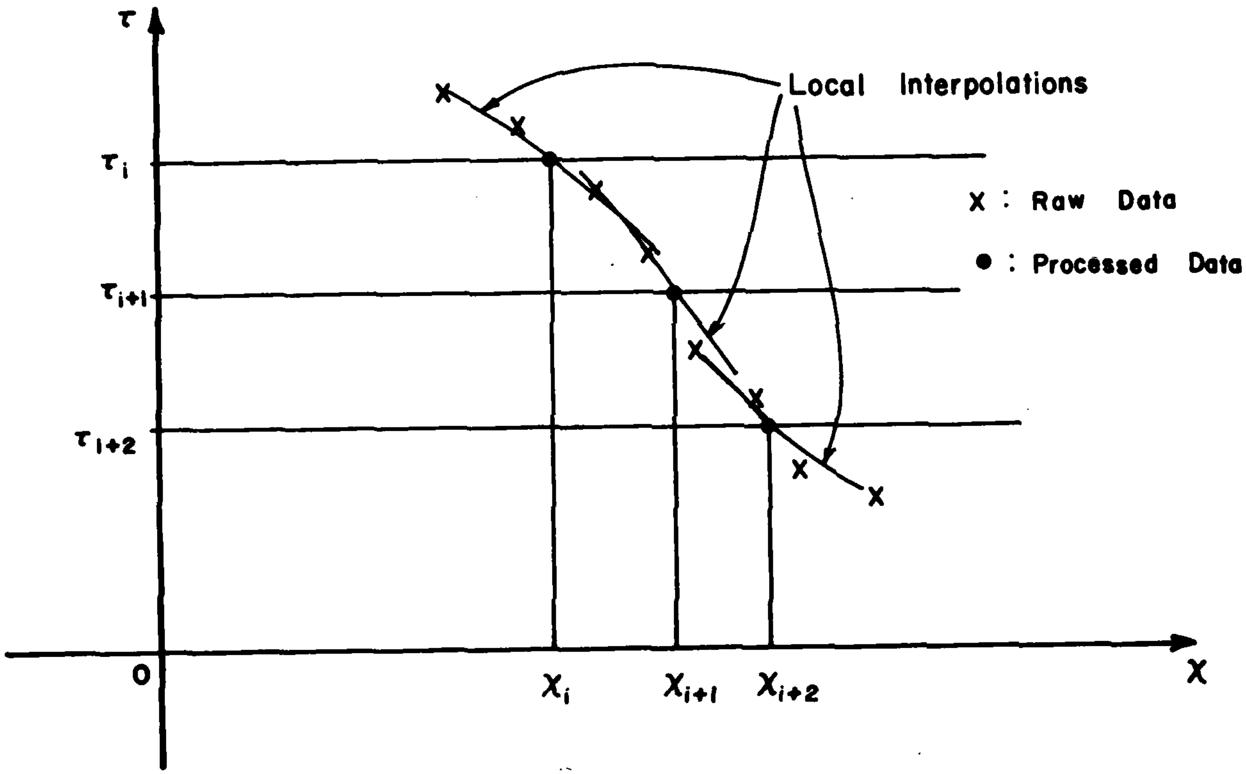


Fig. 3. Pre-processing of Data

(τ_j, x_j), and the derived equal transmission data.

computations. Actual time ranges required for ADSET and SIMMIN computations will be given in a later section.

VI. Lowtran Capabilities and Functions

6.1 Introduction

The Lowtran code consists of a computer model for the calculation of transmittance through atmospheres containing absorbing and scattering molecules and aerosols. The models used in the code were for the most part developed in 1972[6] but later editions incorporated computational changes and other capabilities[7-10]. It covers the spectral range from 0.25 to 28.5 μm at intervals of 5 cm^{-1} with a resolution for the major absorbers of 20 cm^{-1}. The transmittance calculation is made on six model atmospheres and two haze models on a 33-level basis for altitude, pressure, temperature and density from sea-level to 100 km. The path of the transmission is considered to be refracted by changes in atmospheric density, a fact taken into account in an optional subroutine. In its present form the Lowtran code consists of a single main program that inputs the path data and model parameters, computes the equivalent absorber amount, and performs the transmittance calculations. The only present subroutines are associated with the path, and are optional. The difficulties of understanding and, especially, updating such a program structure are considerable.

The principal objective of this effort is to modify the program structure of the Lowtran code in accordance with the following criteria:

1. The basic functions, calculations and print-outs remain nearly identical to the original.

2. The basic operations involving the reading of data, the calculation of the equivalent path and the transmittance calculations are all separate, independent programs, but are connected as subroutines to a main control program.

3. The structure modification is performed on the latest version of the code, i.e., Lowtran 4.

As an exercise in the use of the modularized version, the present authors added empirical band models for transmittance through the trace gases. Also, continuous functions were made to replace their transmission tables for the principal molecular absorbing species.

6.2 General Features

In this section an effort is made to summarize the basic structure, fundamental calculations and models used in the Lowtran code for estimating atmospheric attenuation by gases and aerosols. Reference is specifically made to the latest fourth version, although at the present time the authors are aware of a recent effort by AFGL on a fifth version. From the authors' evaluation of their recent efforts, it appears that the modularization presented here may be incorporated in their latest version. For instance, the latter is known to have a single separate subroutine for the emission and transmission calculations. The major contribution of the work presented here lies in the separation of that emission and transmission loop into a subroutine for model selection, a subroutine for the equivalent path and individual subroutines for all of the attenuation models in the code.

The Lowtran code is designed for the specific purpose of calculating at low resolutions either atmospheric radiance or transmittance between any two locations in the Earth's atmosphere at frequencies ranging from the ultraviolet (UV) to the infrared (IR). This is accomplished through the use of band models accounting for resonant gaseous absorption (e.g. H_2O vapor, O_3, HNO_3 vapor and the uniformly-mixed gases), resonant aerosol absorption, non-resonant gaseous absorption (e.g. N_2 and H_2O vapor continua) and scattering by molecules and aerosols. The spectral intervals over which the band models are provided vary from 5 cm^{-1} to 500 cm^{-1}, as shown in Table 1. It should be pointed out that the spectral resolution is generally much lower than the interval over which they are defined. For instance, the models for the principal absorbers are given at 5 cm^{-1} intervals, while their spectral resolution is 20 cm^{-1}. The spectral resolutions for the remaining models is not specified anywhere in the available literature on the code. In this table the spectral definition of the models for aerosol absorption, and for aerosol and molecular scattering are not shown because they are spectrally continuous.

The spectral regions over which the attenuation models are effective are summarized in Table 2. It may be seen in this table that over some regions only a few species attenuate and, therefore, a transmittance of unity may be specified in the calculation of the total transmittance. This table forms the basis for the model selection subroutine introduced in the modularized version for the purpose of simplifying the code structure.

In the discussion that follows, the individual

ATTENUATING SPECIES	MODEL FREQUENCY INTERVAL (cm^{-1})			
	5	50	200	500
H_2O				
UNIFORMLY-MIXED GASES				
O_3				
N_2 CONTINUUM				
H_2O CONTINUUM				
HNO_3				
VISIBLE O_3				
ULTRA VIOLET O_3				

Table 1. Frequency interval of the attenuation band models in the Lowtran code. The models for aerosol absorption and aerosol and molecular scattering are spectrally continuous and, therefore, not shown.

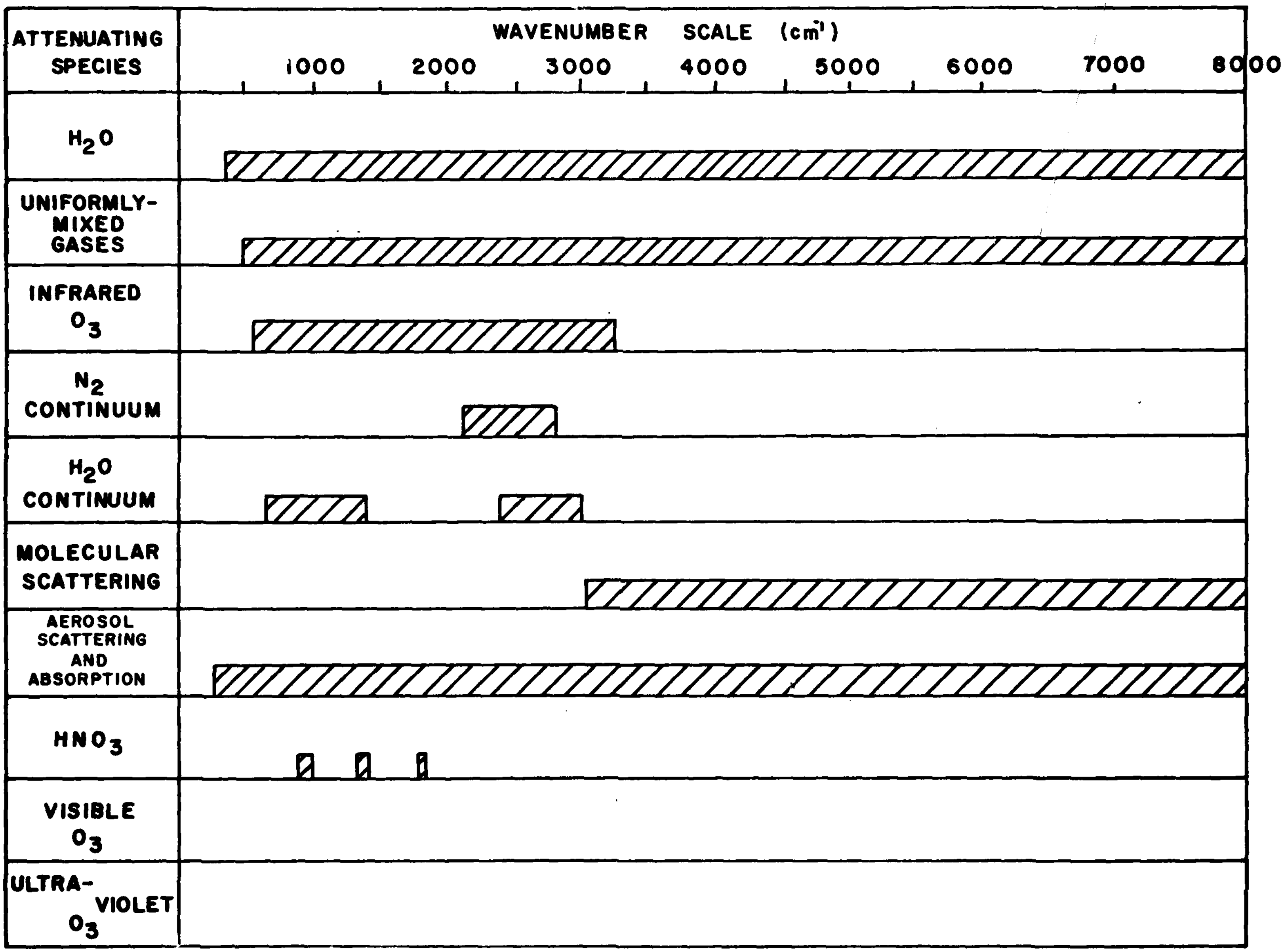


Table 2. Spectral region over which the attenuation models in LOWTRAN are effective.

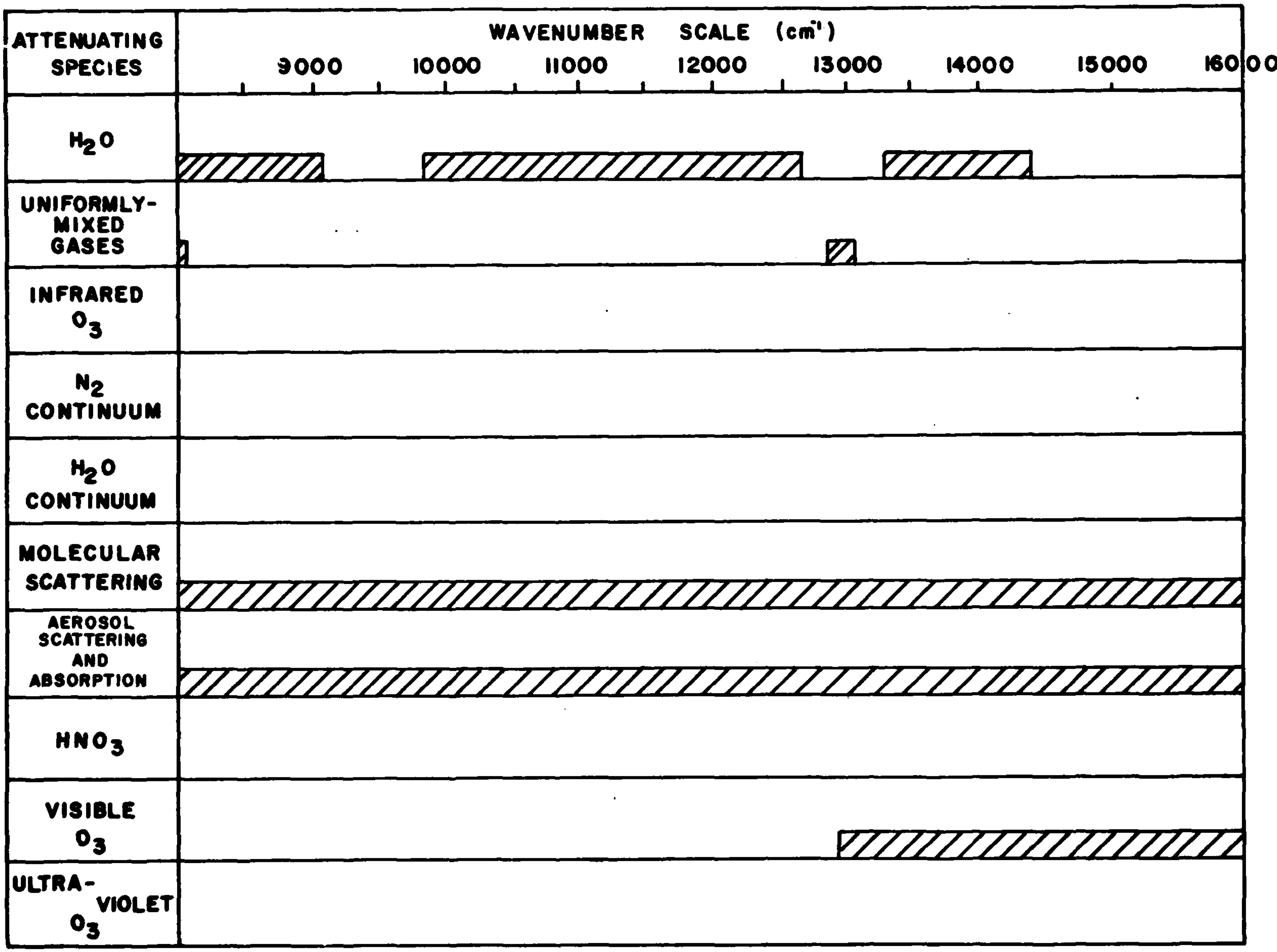


Table 2. (Continued)

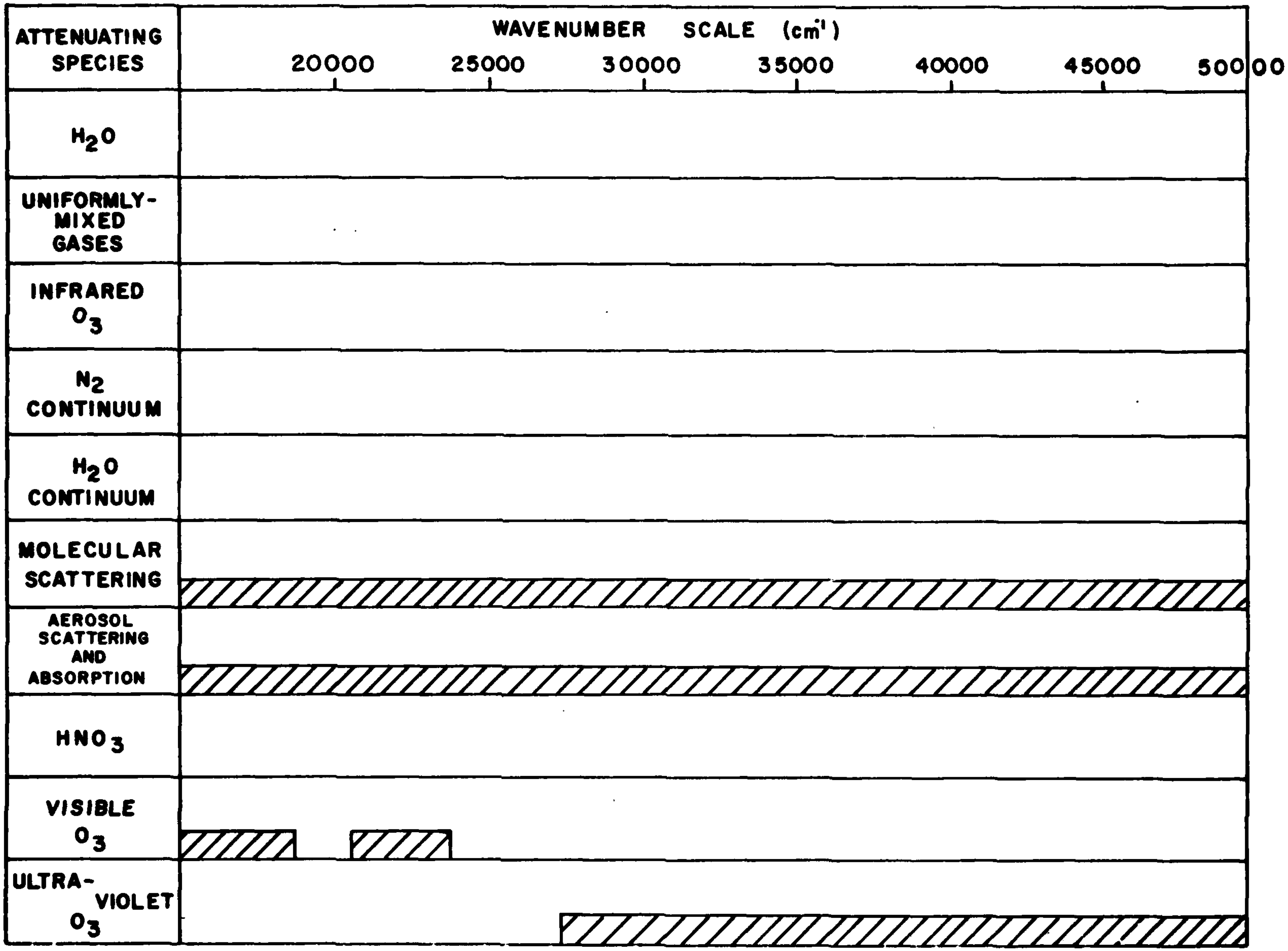


Table 2. (Continued)

attenuation models are grouped together in certain classes and are briefly discussed. Generally speaking, the discussion is restricted to the extent of illustrating the function and parameters which had to be identified in Lowtran for the modularization purpose that followed. An exception is made in the case of the major molecular absorption models (i.e. H_2O vapor, infrared O_3 and the uniformly-mixed gases) because they are replaced with continuous functions in the modularized version. For a comprehensive discussion on the theory of all of the original models the reader is encouraged to study the series of AFGL reports[6-10] on the code, as well as the references therein.

6.3 Resonant Molecular Absorption Models

Molecular resonant absorption is modeled in the code for H_2O vapor, infrared O_3, the uniformly-mixed gases, and HNO_3 vapor. Different approaches are used for the first three listed as compared with the approaches used in connection with O_3 in the visible and ultraviolet regions and with HNO_3 vapor in the infrared.

The models used to account for gaseous absorption by the molecules of H_2O vapor, infrared O_3, and the uniformly-mixed gases are based on Eq.(8), namely

$$\tau = f\{x\}. \tag{8}$$

The developers of Lowtran obtained the parameters n, m, the function f and the spectral constant C' at 5 cm^{-1} intervals using experimental and calculated transmittance data of 20 cm^{-1} resolution. Table 3 shows the values of the parameters, as well as, the equations for the calculation of the absorber amount. The spectral constant C' over the entire spectrum of definition may be found as part of the data input presented in the Appendix. The transmission model for the uniformly-mixed gases was obtained by combining the data for all of these gases in the proportions listed in Table 4. It should be pointed out that the temperature and pressure exponents used in Lowtran for the major absorbers and listed in Table 3 are not the same as the ones developed from the original transmission data. This inconsistency was introduced during the digitizing of the curves for inclusion in the computer code, in order to account more accurately for the temperature dependence[2].

The method used for modeling HNO_3 vapor and the visible and ultraviolet O_3 is similar to the one described above for the major absorbers, except that the function f was specified *a priori* to be an exponential. Thus,

$$\tau = \exp(-CW), \tag{38}$$

where for HNO_3

$$W = \left(\frac{P}{P_o}\right) \left(\frac{T_o}{T}\right) U, \qquad (39)$$

$$U = M\ Z \times 10^5, \qquad (40)$$

and for O_3

$$W = U = 46.667\ \rho Z. \qquad (41)$$

In Eq. (40) M is the mixing ratio profile as tabulated in the Appendix together with the C's, and ρ is the absorber density.

The last of the molecular absorption models is the one for the resonant absorption by atmospheric aerosols. The exponential function in Eq. (38) is assumed

$$\tau = \exp(-CW),$$

where

$$W = U = 3.5336 \times 10^{-6} NZ, \qquad (42)$$

and N is the vertical distribution of the number of haze particles. Tabulations are provided of distributions for 5 Km and 23 km visibility, as listed in the Appendix. Other visibilities are treated in the code itself through linear interpolation.

6.4 Non-Resonant Molecular Absorption Models

Non-resonant gaseous molecular absorption is represented by the N_2 and H_2O vapor continuums. The same modeling approach is used for N_2 as for resonant molecular absorption, that is

$$\tau = \exp(-CW),$$

where

$$W = \left(\frac{P}{P_o}\right)^2 \left(\frac{T_o}{T}\right)^{1.5} U \tag{43}$$

$$U = 0.8\ Z. \tag{44}$$

For the H_2O vapor continuum an exponential function is also used, but with a more elaborate exponent. Thus,

$$\tau = \exp(-\gamma), \tag{45}$$

where

$$\gamma = C_s[P_w + \frac{C_n}{C_s}(P - P_w)]\ U. \tag{46}$$

Here, P_w is the partial pressure of water and C_s and C_n are the self-broadening and nitrogen-broadening spectral constants. The values of these spectral constants depend on the spectral region where the continuum is effective.

In the 8 to 14 μm region

$$C_s = C_o \exp[6.08\ (\frac{296}{T} - 1)], \tag{47}$$

and

$$\frac{C_n}{C_s} = 0.002, \tag{48}$$

while in the 3.5 to 4.2 μm region

$$C_s = C_o \exp[4.56\ (\frac{296}{T} - 1)], \tag{49}$$

and

$$\frac{C_n}{C_s} = 0.120. \tag{50}$$

In these equations the value of C_o is given by

$$C_o = 4.18 + 5578 \exp(-7.87 \times 10^{-3}\nu). \tag{51}$$

ATTENUATING SPECIE	SPECTRAL REGION (cm^{-1})	PRESSURE EXPONENT n	TEMPERATURE EXPONENT m	ABSORBER AMOUNT U
H_2O Vapor	350- 9,195 9,875-12,795 13,400-14,520	0.90	0.45	0.1 ρZ
Uniformly-Mixed Gases	500- 8,070 12,950-13,245	1.75	1.375	Z
Infrared O_3	575- 3,270	0.40	0.20	46.667 ρZ
N_2 Continuum	2,080- 2,740	2.00	1.50	0.8 Z
Aerosol Absorption	333-50,000	0.00	0.00	3.5336×10^{-6} NZ
Aerosol Scattering	333-50,000	0.00	0.00	3.5336×10^{-4} NZ
Molecular Scattering	3,000-50,000	1.00	1.00	9.87×10^{-20} Z
HNO_3 Vapor	850- 920 1,275- 1,350 1,675- 1,735	1.00	1.00	1×10^5 MZ
Visible and Ultraviolet O_3	13,000-24,000 27,500-50,000	0.00	0.00	46.667 ρZ

Table 3. Absorber parameters in Lowtran for the attenuation models, where ρ is the density, Z the range and M the mixing ratio. The H_2O continuum model is excluded because of its different functional form.

GAS	MOLECULAR WEIGHT	PARTS PER MILLION BY VOLUME (ppm)
CO_2	44	330.0
N_2O	44	0.28
CO	28	0.075
CH_4	16	1.60
O_2	32	2.095×10^5

Table 4. Concentrations of the uniformly-mixed gases used in the combined model.

6.5 Scattering Models

In order to account for atmospheric scattering exponential functions were used again. For scattering by molecules the model is defined as in Eq. (38)

$$\tau = \exp(-CW)$$

where

$$W = \left(\frac{P}{P_o}\right)\left(\frac{T_o}{T}\right) U \tag{52}$$

$$U = 9.87 \times 10^{-20}\ Z \tag{53}$$

$$C = \nu^4 \tag{54}$$

For aerosol scattering the argument of the assumed exponential function is

$$W = U = 3.5336 \times 10^{-4}\ NZ \tag{55}$$

VII. Modularization of Lowtran Including the Trace Gases

7.1 Introduction

Considering the generality and broadness in scope of this code it is not surprising that the program structure shows in its present form great complexity. Although the program user is not normally interested in aspects of the code other than the input and output, there are many cases where a basic understanding helps in specific applications. Situations are likely to occur, for instance, where a replacement of one of the several attenuation models is highly desirable. To assist in the implementation of model additions or changes as well as in the extension to other spectral regions and media, the concept of the modularized version was conceived. This version[15] was designed to represent exactly the same calculations as the original, except for the simplification of the program structure into modules or subroutines. However, upon the termination of that task the authors proceeded to add models for the trace gases, as developed during the present scientific effort.

7.2 Structure of Modularized Version

The basic design used was that of a main program which reads input data, computes total transmittance and radiance and generates outputs, and a series of subroutines which select individual models and compute individual transmittances and absorber amounts. This is shown in Fig. 4. The main operational flow chart follows in Fig. 5. Excluding the four subroutines for the trace gases, the modularized version breaks down the original into one program with 11 subroutines. The flow chart for subroutine ABSORB is shown in Fig. 6. This subroutine computes the equivalent absorber amount for all of the attenuation models according to Eq.(4), which in terms of the meteorological variables becomes

$$W = \int \left(\frac{P(Z)}{P_o}\right)^n \left(\frac{T_o}{T(Z)}\right)^m dU \qquad (56)$$

Figure 7 gives details of the Transmittance/Radiance Loop of program Main. It is worth noting that the modularized version of Lowtran being done by AFGL separates this loop into a subprogram. The modularization discussed in this text leaves the loop as part of the main program, but extracts individual subroutines for the calculation of the equivalent absorber amount, the frequency selection, and the attenuation models.

The flow chart for FREQSL subroutine is shown in Fig. 8. This subroutine is designed to simplify the process of arriving at the individual models effective at the frequency of interest. It should also assist the

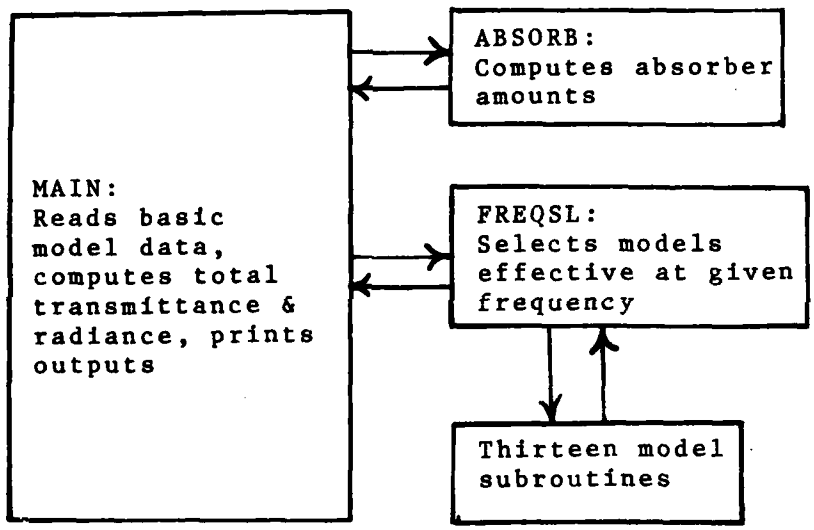


Fig. 4. Conceptual flow chart of modularized Lowtran.

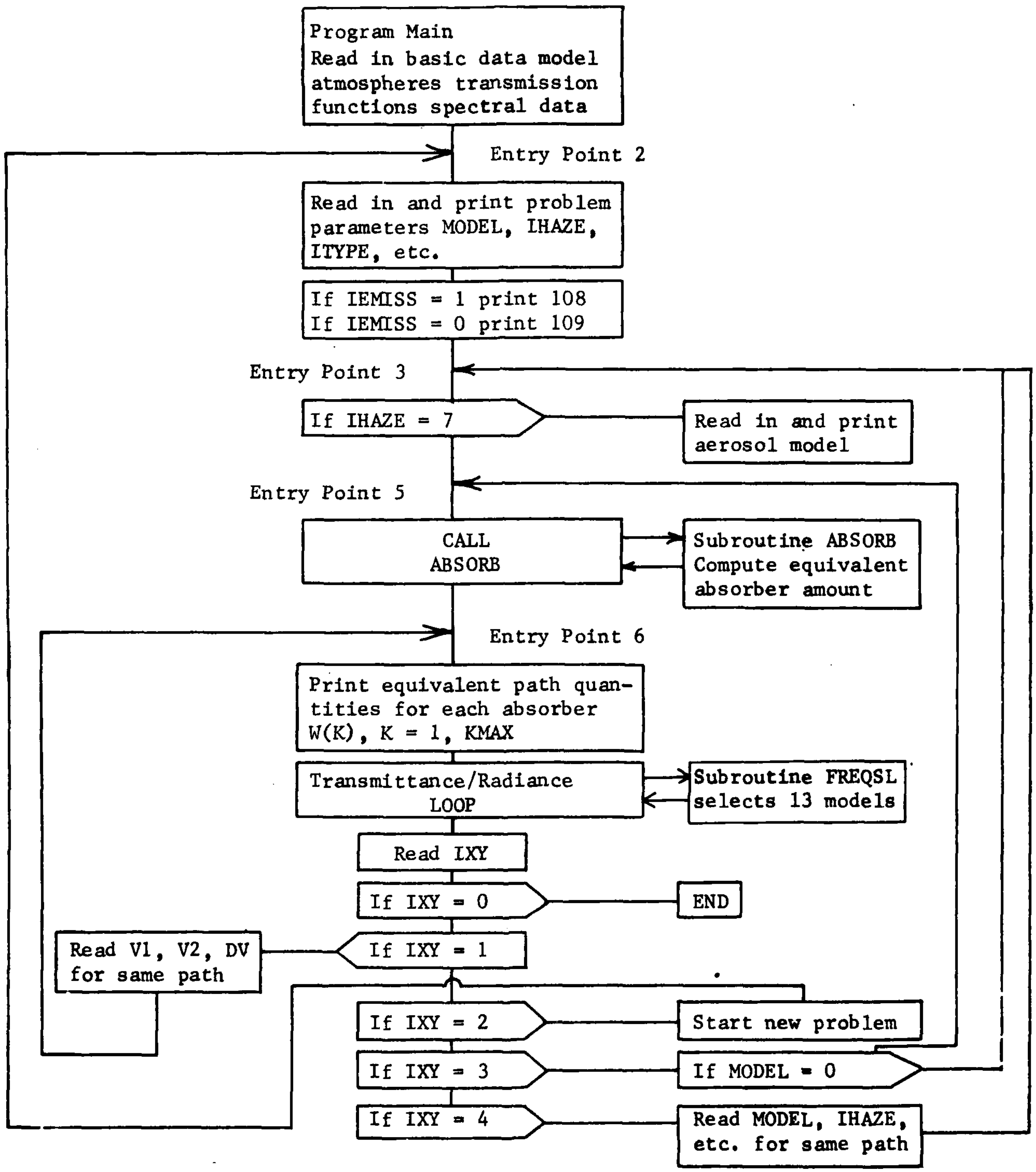


Fig. 5. General flow chart for Modularized Lowtran 4

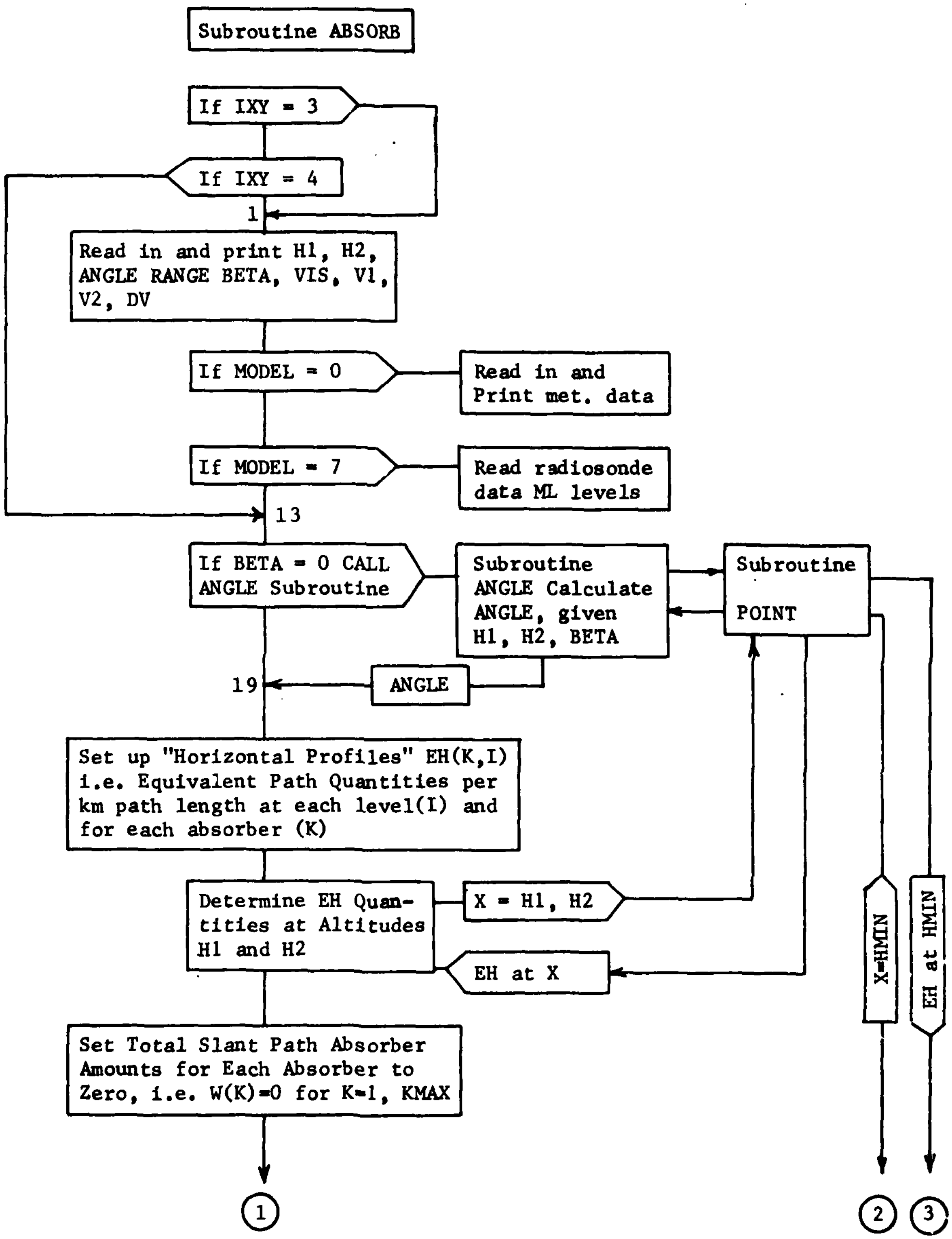


Fig. 6. Flow chart for subroutine ABSORB, computing equivalent path quantities

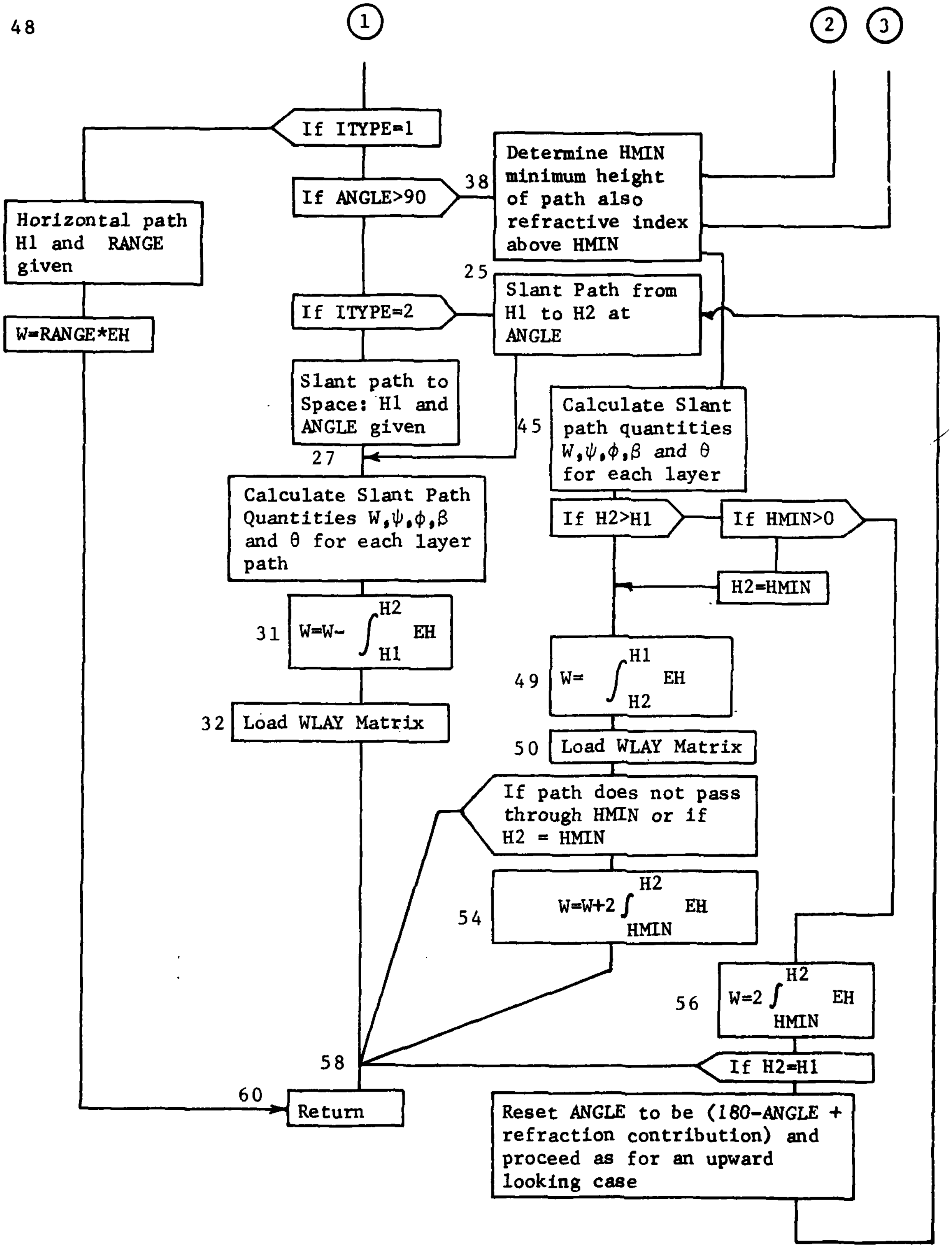

1
2
3
If ITYPE=1
Horizontal path H1 and RANGE given
W=RANGE*EH
If ANGLE>90
38
Determine HMIN minimum height of path also refractive index above HMIN
25
If ITYPE=2
Slant Path from H1 to H2 at ANGLE
Slant path to Space: H1 and ANGLE given
45
Calculate Slant path quantities W,ψ,φ,β and θ for each layer
27
Calculate Slant Path Quantities W,ψ,φ,β and θ for each layer path
If H2>H1
If HMIN>0
H2=HMIN
31
W=W− ∫ H1 H2 EH
49
W= ∫ H2 H1 EH
32
Load WLAY Matrix
50
Load WLAY Matrix
If path does not pass through HMIN or if H2 = HMIN
54
W=W+2 ∫ HMIN H2 EH
56
W=2 ∫ HMIN H2 EH
58
If H2=H1
60
Return
Reset ANGLE to be (180-ANGLE + refraction contribution) and proceed as for an upward looking case

Fig. 6. (cont'd)

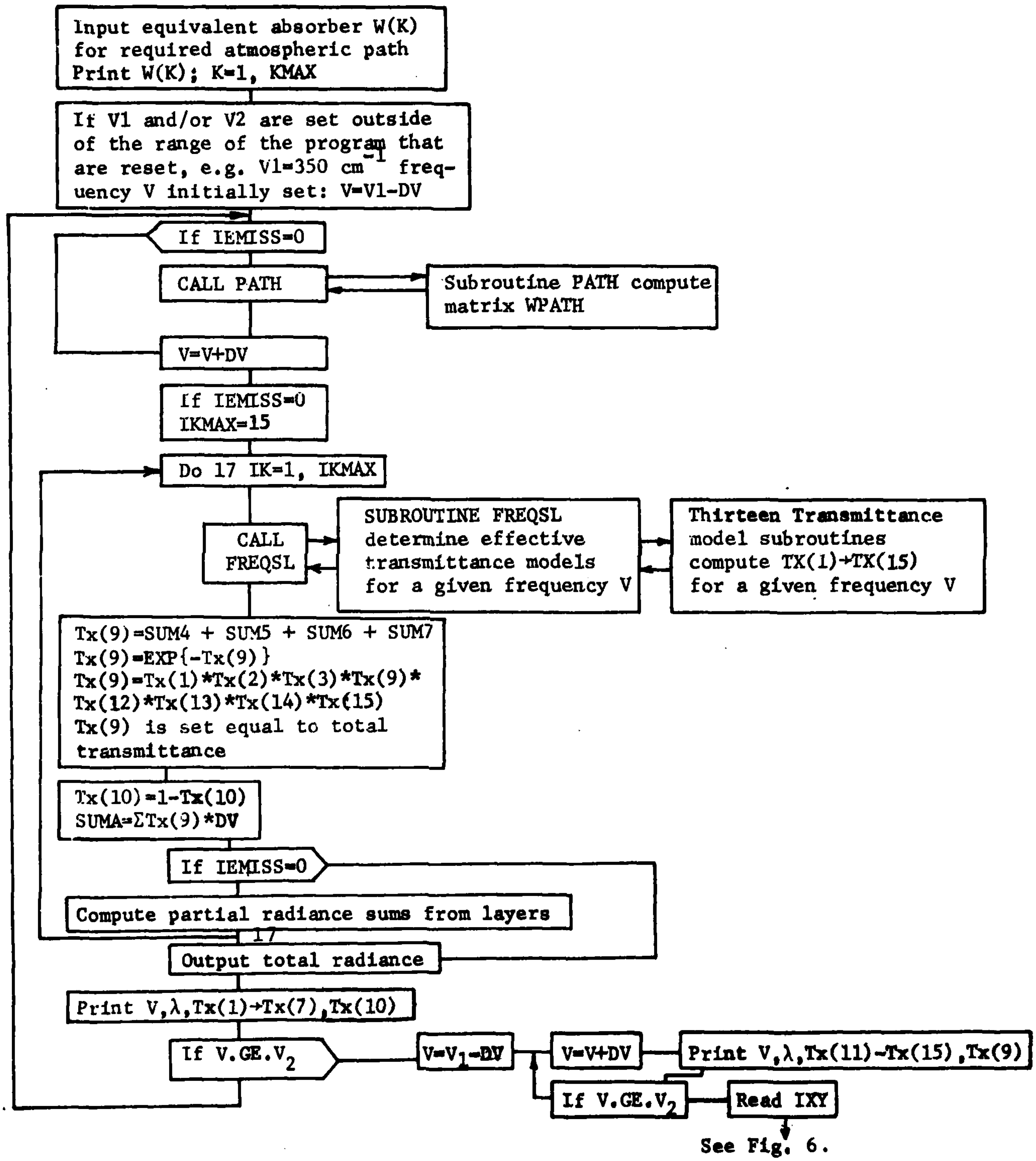


Fig. 7. Flow chart for transmittance/radiance loop.

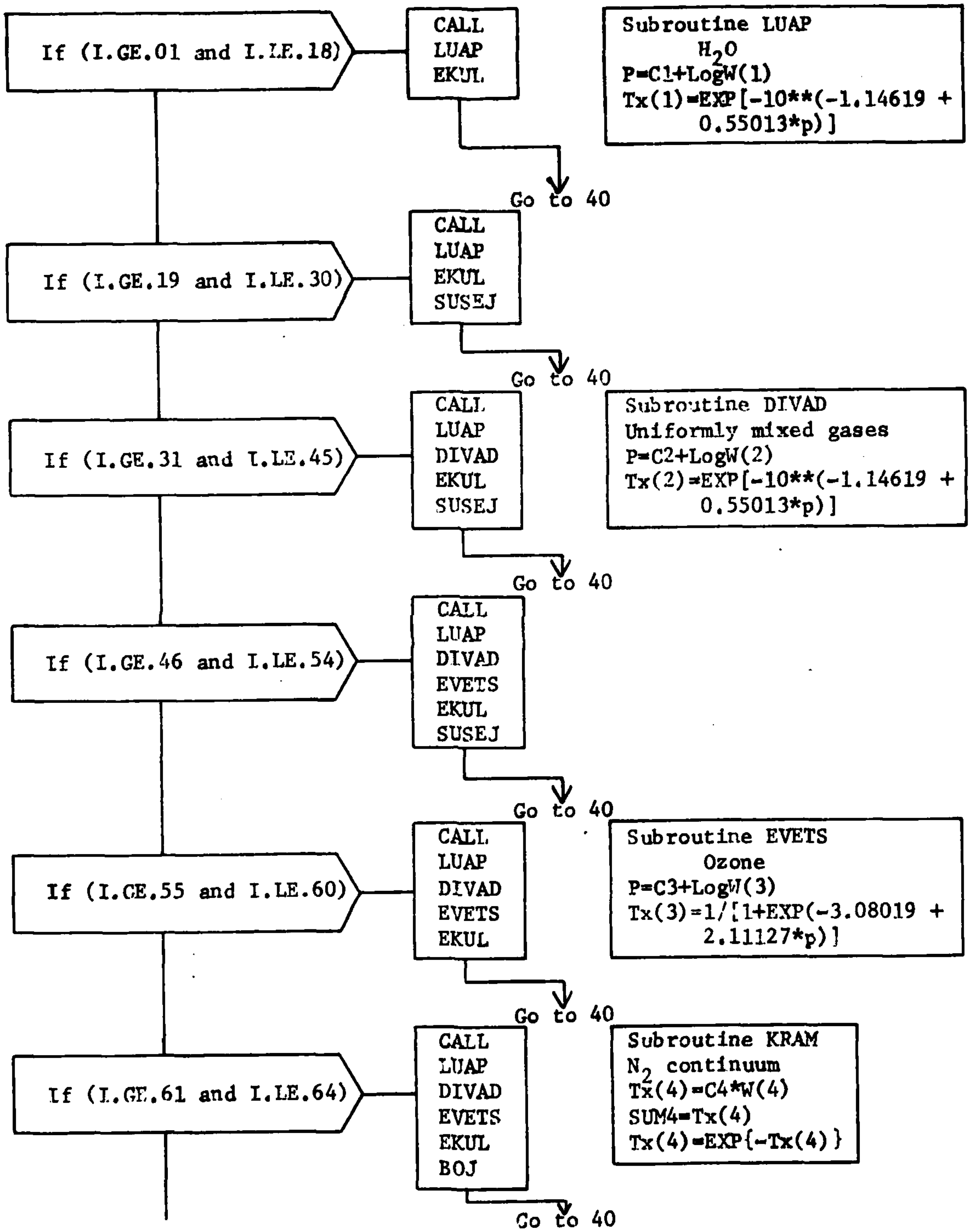


Fig. 8. Flow chart for subroutine FREQSL and thirteen transmittance model subroutines.

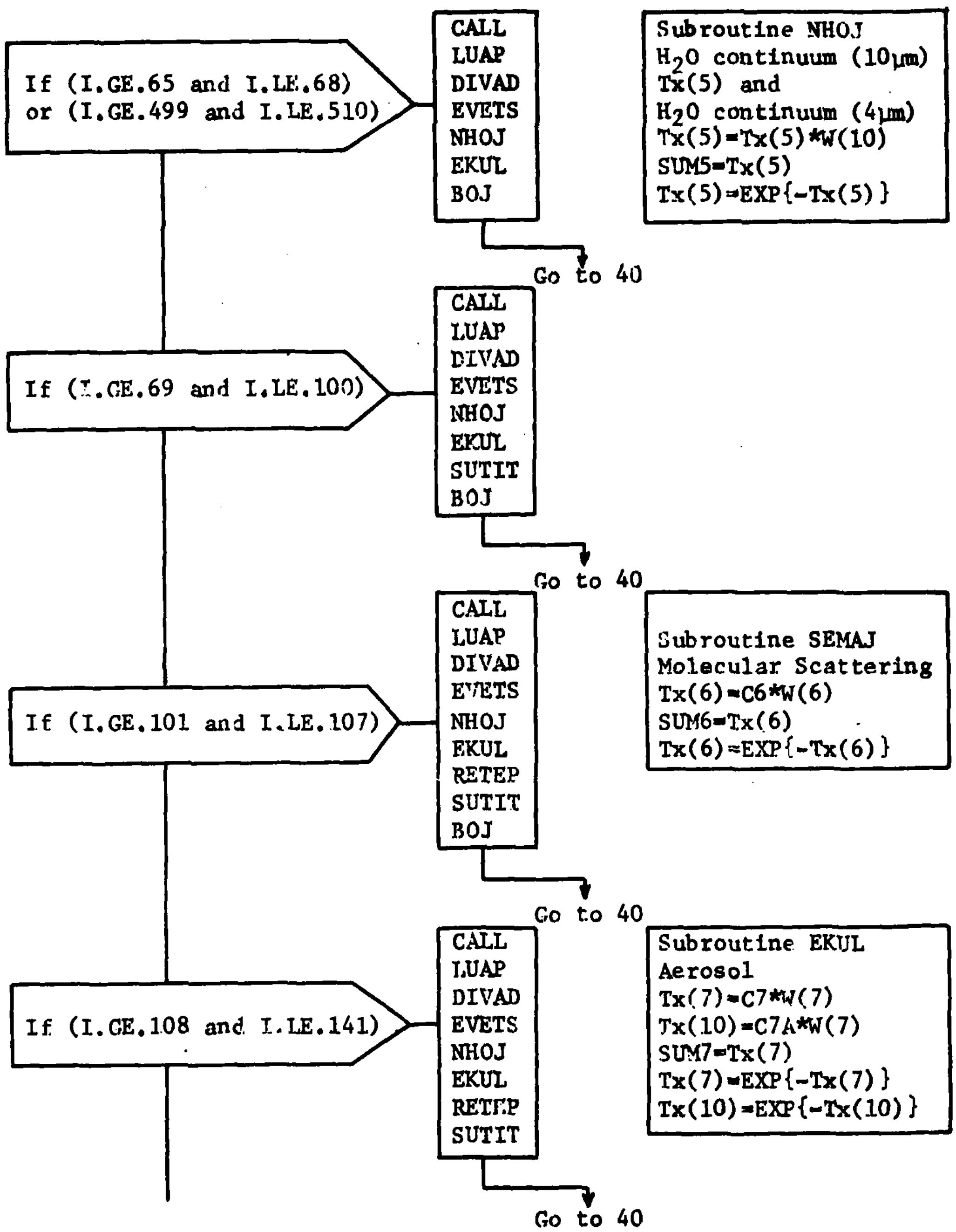


Fig. 8. Continued

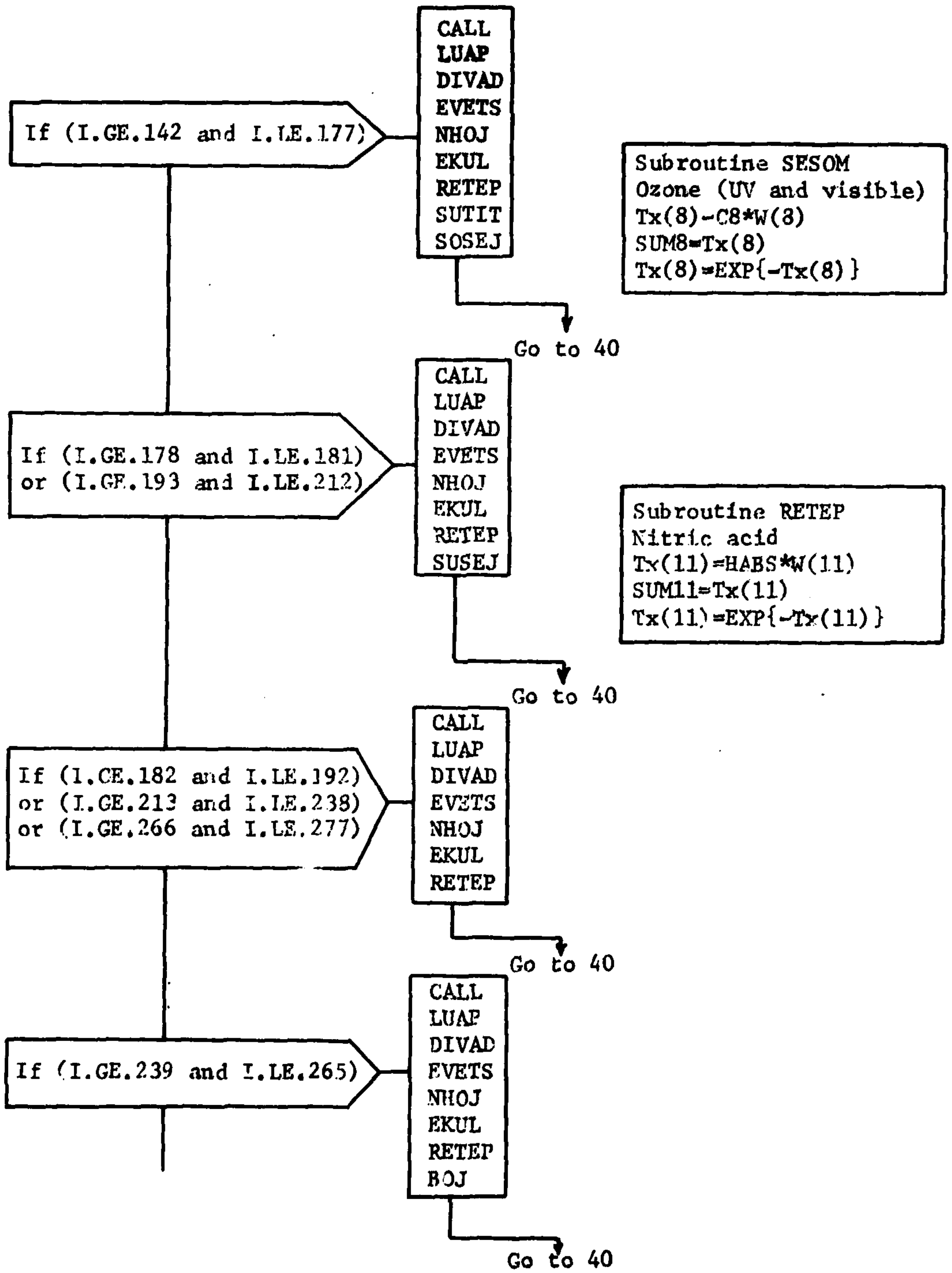


Fig. 8. Continued

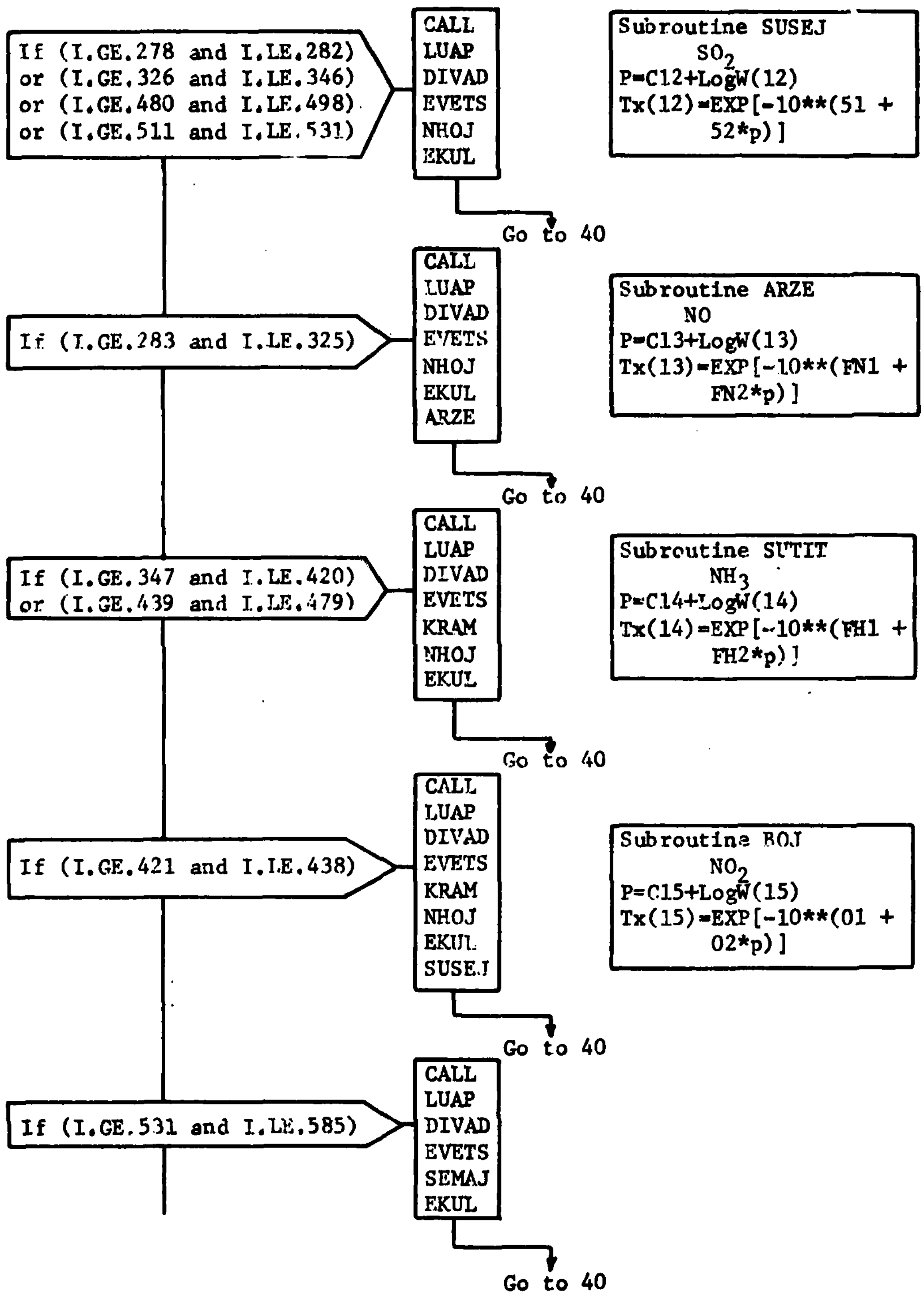


Fig. 8. Continued

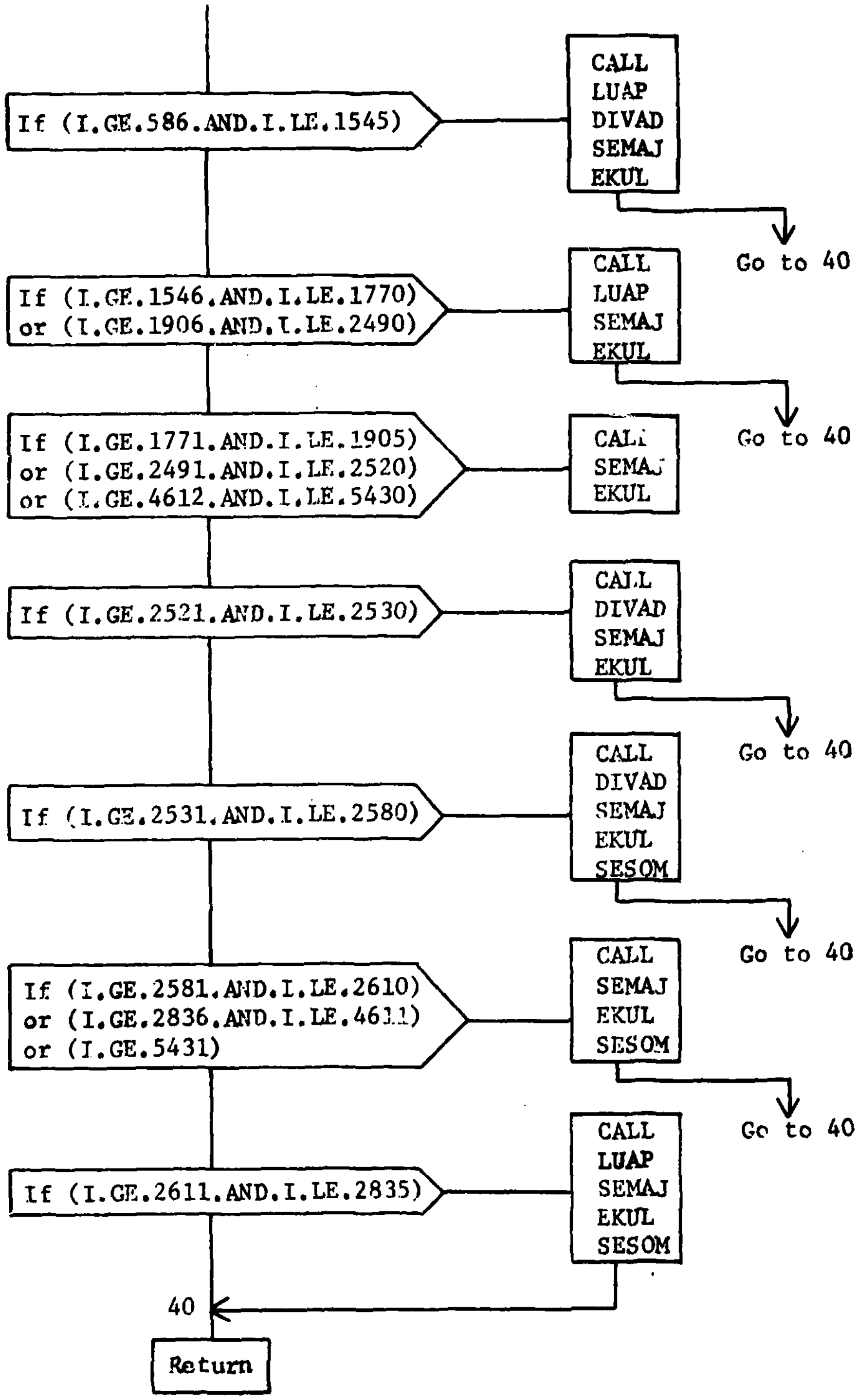


Fig. 8. Continued

user who desires to replace or add models to the program and it should reduce the overall computational time. This subroutine is based on Tables 2 and 8.

7.3 Models for H_2O Vapor, Infrared O_3 and the Uniformly-Mixed Gases

As indicated above, all the attenuation models were extracted from the main program and placed into subroutines. The models were left basically in the same structural form except for the models for HNO_3 and H_2O vapor, infrared O_3 and the uniformly-mixed gases. The change in the first was to arrange it along the same form as in the other models originally available in Lowtran. That is, the spectral parameters were extracted from the subroutine and read at the beginning of program MAIN. The changes in the latter three gases (i.e. H_2O vapor, O_3 in the infrared and the uniformly-mixed gases) were based on a previous work by Pierluissi et al.[2] on the representation of the tabulated transmission functions by analytical functions. The other principal change consisted of adding models for the trace gases SO_2, NO, NO_2 and NH_3.

To arrive at the analytical function for modeling H_2O vapor and the uniformly-mixed gases the double exponential expression

$$\tau = \exp(-10^{a_o + a_1 x}) \qquad (57)$$

where x is as in Eq. 9 and a_o and a_1 are absorber constants, was curve-fitted to the 134 values of τ and x tabulated in Lowtran. The values found are $a_o = -1.14619$ and $a_1 = 0.55013$, and it reproduced the tabulated transmittance with a standard deviation of 0.005. For O_3 the function adopted is given by

$$\tau = \frac{1}{1 + e^{a_o + a_1 x}} \tag{58}$$

where $a_o = -3.08019$, $a_1 = 2.11127$, and the tabulated data is reproduced with a standard deviation of 0.007. Note in each one of these functions that the 134 tabulated values are replaced with two and, hence, their adoption reduces the computer storage requirements. Also, they inherently offer exponential interpolation while with the present tabulation linear interpolation is being used. Finally, there is no need for the small optical thickness (i.e. $0.999 \leq \tau \leq 1$) correction inserted in Lowtran 4, as required by its radiance calculational scheme.

7.4 Models for Trace Gases SO_2, NO, NO_2, and NH_3

Absorption by the trace gases was incorporated in Lowtran using a somewhat similar procedure. Empirical transmission functions were first obtained from a computerized procedure which replaced the classical manual graphical techniques. The procedure is explained in Chapter III of this report and has been proposed to the scientific community[11].

Instead of either representing the transmission function by a table or by a single function, it was divided into nine segments for each absorber. The individual curve segments are summarized in Table 5, each one being represented by the function

$$\tau = \exp(-10^{a_0 + a_1 x}) \tag{59}$$

For each absorber x is computed with Eqs. (8) through (11) and the relation

$$U = 0.772 \times 10^{-4} \text{ ppm } \rho_a Z \tag{60}$$

where ppm is the parts per million by volume, ρ_a is the air density in gm/m^3 and Z is the range in kilometers. Table 6 lists the ppm and temperature and pressure exponents used in the modularized code for the individual trace gases. The ppm values are read as input through a separate card which may be easily changed according to the needs of the user. The constants C' are tabulated in Table 7. The spectral coverage for each gas is depicted in Table 8. The models are for a resolution of 20 cm^{-1} and are defined at 5 cm^{-1} through their spectral regions of effectiveness. Their mean standard deviation in fitting the original line-by-line data is about 0.008. Figure 9 depicts the transmission functions for the four trace gases considered.

CURVE SEGMENT	TRANSMITTANCE INTERVAL	x-INTERVAL	FUNCTION CONSTANTS	
			a_0	a_1
1	1.000 ~ 0.900	$x \leq -1.057$	0.0682	0.9894
2	0.900 ~ 0.800	-1.057 ~ -0.725	0.0594	0.9811
3	0.800 ~ 0.700	-0.725 ~ -0.514	0.0492	0.9670
4	0.700 ~ 0.600	-0.514 ~ -0.350	0.0408	0.9506
5	0.600 ~ 0.500	-0.350 ~ -0.208	0.0343	0.9319
6	0.500 ~ 0.400	-0.208 ~ -0.074	0.0295	0.9091
7	0.400 ~ 0.300	-0.074 ~ 0.061	0.0273	0.8792
8	0.300 ~ 0.200	0.061 ~ 0.212	0.0300	0.8353
9	0.200 ~ 0.0	$x \geq 0.212$	0.0466	0.7568

Table 5a. Constants for the curve segments in the empirical transmission function for SO_2.

CURVE SEGMENT	TRANSMITTANCE INTERVAL	x-INTERVAL	FUNCTION CONSTANTS	
			a_0	a_1
1	1.000 ~ 0.900	$x \leq -1.158$	-0.0228	0.8240
2	0.900 ~ 0.800	-1.158 ~ -0.684	-0.1822	0.6864
3	0.800 ~ 0.700	-0.684 ~ -0.333	-0.2537	0.5818
4	0.700 ~ 0.600	-0.333 ~ -0.047	-0.2660	0.5450
5	0.600 ~ 0.500	-0.047 ~ 0.199	-0.2663	0.5388
6	0.500 ~ 0.400	0.199 ~ 0.419	-0.2685	0.5497
7	0.400 ~ 0.300	0.419 ~ 0.626	-0.2785	0.5737
8	0.300 ~ 0.200	0.626 ~ 0.833	-0.3000	0.6080
9	0.200 ~ 0.0	$x \geq 0.833$	-0.3373	0.6528

Table 5b. Constants for the curve segments in the empirical transmission function for NO.

CURVE SEGMENT	TRANSMITTANCE INTERVAL	x-INTERVAL	FUNCTION CONSTANTS	
			a_0	a_1
1	1.000 ~ 0.900	$x \leq 0.215$	-1.1877	0.9771
2	0.900 ~ 0.800	0.215 ~ 0.556	-1.1835	0.9577
3	0.800 ~ 0.700	0.556 ~ 0.775	-1.1668	0.9277
4	0.700 ~ 0.600	0.775 ~ 0.949	-1.1416	0.8952
5	0.600 ~ 0.500	0.949 ~ 1.104	-1.1063	0.8580
6	0.500 ~ 0.400	1.104 ~ 1.252	-1.0615	0.8174
7	0.400 ~ 0.300	1.252 ~ 1.406	-1.0055	0.7727
8	0.300 ~ 0.200	1.406 ~ 1.579	-0.9400	0.7260
9	0.200 ~ 0.0	$x \geq 1.579$	-0.8683	0.6807

Table 5c. Constants for the curve segments in the empirical transmission function for NO_2.

CURVE SEGMENT	TRANSMITTANCE INTERVAL	x - INTERVAL	FUNCTION CONSTANTS	
			a_0	a_1
1	1.000 ~ 0.900	$x \leq -1.444$	0.2775	0.8692
2	0.900 ~ 0.800	-1.444 ~ -1.005	0.0962	0.7436
3	0.800 ~ 0.700	-1.005 ~ -0.661	-0.0570	0.5913
4	0.700 ~ 0.600	-0.661 ~ -0.340	-0.1261	0.4867
5	0.600 ~ 0.500	-0.340 ~ -0.033	-0.1450	0.4312
6	0.500 ~ 0.400	-0.033 ~ 0.267	-0.1459	0.4037
7	0.400 ~ 0.300	0.267 ~ 0.575	-0.1409	0.3852
8	0.300 ~ 0.200	0.575 ~ 0.921	-0.1290	0.3645
9	0.200 ~ 0.0	$x \geq 0.921$	-0.1224	0.3573

Table 5d. Constants for the curve segments in the empirical transmission function for NH_3.

TRACE GAS	SPECTRAL REGION (cm^{-1})	PRESSURE EXPONENT n	TEMPERATURE EXPONENT m	PARTS PER MILLION BY VOLUME ppm
SO_2	440- 615 1,055-1,250 1,310-1,410	0.07122	0.06159	0.221
NO	1,760-1,970	0.90098	1.01192	0.250
NO_2	655- 880 1,540-1,670 2,840-2,895	0.18066	0.20911	0.090
NH_3	670-1,230	0.52125	-0.60438	0.200

Table 6. Absorber parameters in Modularized Lowtran used with the models for the trace gases.

WAVENUMBER (cm^{-1})	C'	WAVENUMBER (cm^{-1})	C'	WAVENUMBER (cm^{-1})	C'
440	-2.987	1070	-1.653	1320	-1.237
445	-2.330	1075	-1.443	1325	-0.494
450	-1.791	1080	-1.252	1330	0.139
455	-1.370	1085	-1.080	1335	0.613
460	-1.041	1090	-0.926	1340	0.899
465	-0.795	1095	-0.787	1345	1.043
470	-0.613	1100	-0.661	1350	1.090
475	-0.469	1105	-0.544	1355	1.097
480	-0.346	1110	-0.434	1360	1.104
485	-0.233	1115	-0.329	1365	1.093
490	-0.126	1120	-0.230	1370	1.118
495	-0.037	1125	-0.139	1375	1.088
500	0.0	1130	-0.073	1380	0.926
505	-0.008	1135	-0.047	1385	0.534
510	-0.052	1140	-0.057	1390	-0.067
515	-0.102	1145	-0.083	1395	-0.804
520	-0.102	1150	-0.098	1400	-0.768
525	-0.044	1155	-0.071	1405	-1.687
530	0.013	1160	-0.020	1410	-2.469
535	0.039	1165	0.014	2450	-3.669
540	0.014	1170	0.011	2455	-2.855
545	-0.056	1175	-0.040	2460	-2.131
550	-0.141	1180	-0.123	2465	-1.528

Table 7a. The spectral coefficient $C'(\nu)$ for SO_2.

WAVENUMBER (cm^{-1})	C'	WAVENUMBER (cm^{-1})	C'	WAVENUMBER (cm^{-1})	C'
555	-0.221	1185	-0.213	2470	-1.076
560	-0.294	1190	-0.301	2475	-0.805
565	-0.366	1195	-0.388	2480	-0.647
570	-0.442	1200	-0.481	2485	-0.571
575	-0.529	1205	-0.586	2490	-0.549
580	-0.635	1210	-0.707	2495	-0.539
585	-0.766	1215	-0.843	2500	-0.536
590	-0.934	1220	-0.996	2505	-0.517
595	-1.157	1225	-1.165	2510	-0.528
600	-1.457	1230	-1.351	2515	-0.691
605	-1.862	1235	-1.554	2520	-1.073
610	-2.420	1240	-1.777	2525	-1.673
615	-3.094	1245	-2.033	2530	-2.414
1055	-2.604	1250	-2.369	2535	-2.207
1060	-2.156	1310	-3.010		
1065	-1.884	1315	-2.080		

Table 7a. (Continued)

WAVENUMBER	C'	WAVENUMBER	C'	WAVENUMBER	C'
1760	-2.691	1835	-0.231	1910	0.003
1765	-2.521	1840	-0.176	1915	-0.032
1770	-2.328	1845	-0.144	1920	-0.105
1775	-2.115	1850	-0.143	1925	-0.211
1780	-1.894	1855	-0.188	1930	-0.352
1785	-1.685	1860	-0.244	1935	-0.529
1790	-1.485	1865	-0.342	1940	-0.742
1795	-1.296	1870	-0.434	1945	-0.992
1800	-1.117	1875	-0.471	1950	-1.282
1805	-0.947	1880	-0.483	1955	-1.610
1810	-0.792	1885	-0.392	1960	-1.975
1815	-0.649	1890	-0.266	1965	-2.374
1820	-0.519	1895	-0.151	1970	-2.806
1825	-0.407	1900	-0.046		
1830	-0.311	1905	-0.001		

Table 7b. The spectral coefficient $C'(\nu)$ for NO.

WAVENUMBER	C'	WAVENUMBER	C'	WAVENUMBER	C'
655	-0.844	800	-0.255	1,600	2.616
660	-0.760	805	-0.286	1,605	2.616
665	-0.676	810	-0.315	1,610	2.606
670	-0.608	815	-0.334	1,615	2.608
675	-0.543	820	-0.352	1,620	2.643
680	-0.496	825	-0.366	1,625	2.682
685	-0.450	830	-0.396	1,630	2.672
690	-0.414	835	-0.423	1,635	2.576
695	-0.383	840	-0.459	1,640	2.350
700	-0.326	845	-0.498	1,645	1.955
705	-0.289	850	-0.541	1,650	1.346
710	-0.217	855	-0.586	1,655	0.596
715	-0.140	860	-0.630	1,660	-0.258
720	-0.097	865	-0.676	1,665	-1.214
725	-0.034	870	-0.720	1,670	-1.951
730	-0.031	875	-0.766	2,840	-1.220
735	-0.082	880	-0.809	2,845	-0.644
740	-0.139	1,540	-2.428	2,850	-0.253
745	-0.216	1,545	-1.494	2,855	0.052
750	-0.249	1,550	-0.647	2,860	0.326
755	-0.207	1,555	0.122	2,865	0.574
760	-0.117	1,560	0.756	2,870	0.792

Table 7c. The spectral coefficient $C'(\nu)$ for NO_2.

WAVENUMBER	C'	WAVENUMBER	C'	WAVENUMBER	C'
765	-0.047	1,565	1.230	2,875	0.978
770	0.000	1,570	1.568	2,880	1.122
775	0.009	1,575	1.855	2,885	1.216
780	-0.046	1,580	2.104	2,890	1.252
785	-0.100	1,585	2.310	2,895	1.249
790	-0.148	1,590	2.469		
795	-0.214	1,595	2.573		

Table 7c. (Continued)

WAVENUMBER	C'	WAVENUMBER	C'	WAVENUMBER	C'
690	-2.603	875	-1.124	1,060	-0.589
695	-2.456	880	-1.155	1,065	-0.565
700	-2.290	885	-1.161	1,070	-0.537
705	-2.128	890	-1.143	1,075	-0.510
710	-1.980	895	-1.139	1,080	-0.512
715	-2.225	900	-1.117	1,085	-0.528
720	-1.823	905	-1.107	1,090	-0.575
725	-1.744	910	-0.844	1,095	-0.625
730	-1.674	915	-0.558	1,100	-0.668
735	-1.577	920	-0.238	1,105	-0.694
740	-1.481	925	-0.042	1,110	-0.717
745	-1.372	930	-0.002	1,115	-0.740
750	-1.284	935	-0.157	1,120	-0.774
755	-1.207	940	-0.436	1,125	-0.834
760	-1.128	945	-0.610	1,130	-0.905
765	-1.061	950	-0.548	1,135	-0.977
770	-1.004	955	-0.352	1,140	-1.042
775	-0.947	960	-0.139	1,145	-1.133
780	-0.886	965	-0.095	1,150	-1.219
785	-0.876	970	-0.365	1,155	-1.301
790	-0.872	975	-0.729	1,160	-1.383
795	-0.869	980	-1.048	1,165	-1.488
800	-0.872	985	-1.275	1,170	-1.594

Table 7d. The spectral coefficient $C'(\nu)$ for NH_3.

WAVENUMBER	C'	WAVENUMBER	C'	WAVENUMBER	C'
805	-0.848	990	-1.257	1,175	-1.696
810	-0.811	995	-1.142	1,180	-1.796
815	-0.772	1,000	-1.053	1,185	-1.873
820	-0.773	1,005	-0.963	1,190	-1.936
825	-0.793	1,010	-0.920	1,195	-1.991
830	-0.825	1,015	-0.944	1,200	-2.080
835	-0.869	1,020	-0.889	1,205	-2.183
840	-0.894	1,025	-0.829	1,210	-2.292
845	-0.890	1,030	-0.736	1,215	-2.404
850	-0.873	1,035	-0.644	1,220	-2.529
855	-0.868	1,040	-0.596	1,225	-2.639
860	-0.907	1,045	-0.569	1,230	-2.732
865	-0.965	1,050	-0.572		
870	-1.045	1,055	-0.590		

Table 7d. (Continued)

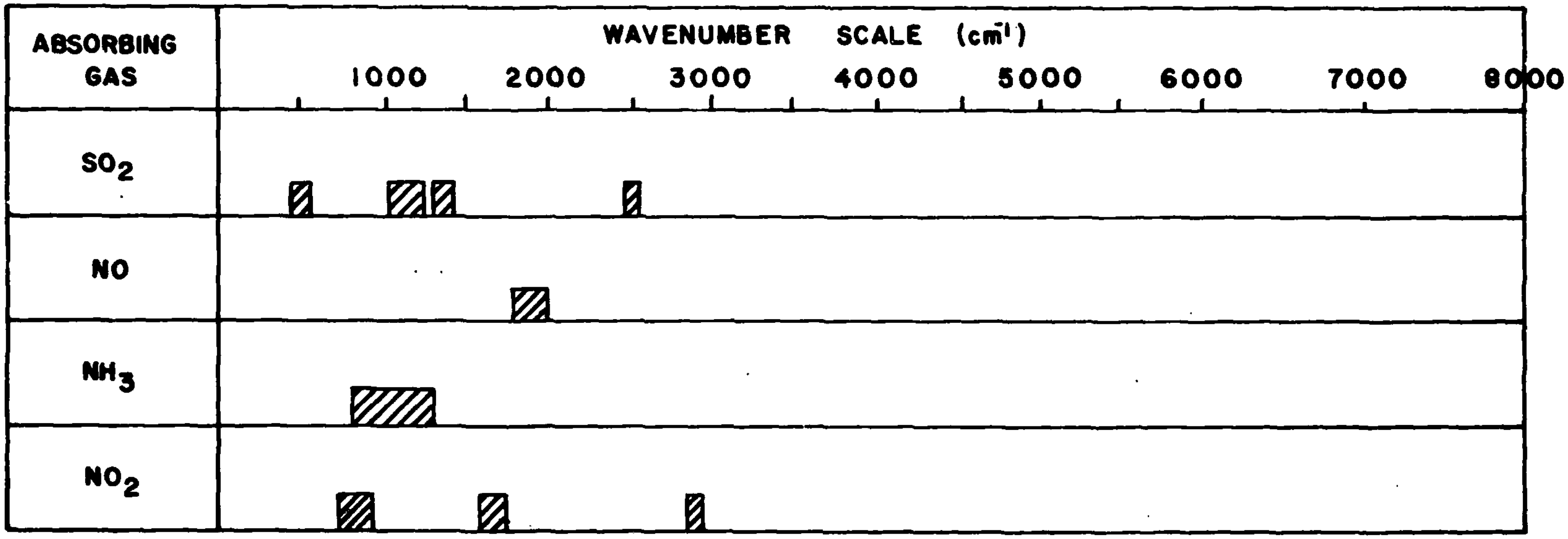


Table 8. Absorption frequency region of the trace gases in the atmosphere.

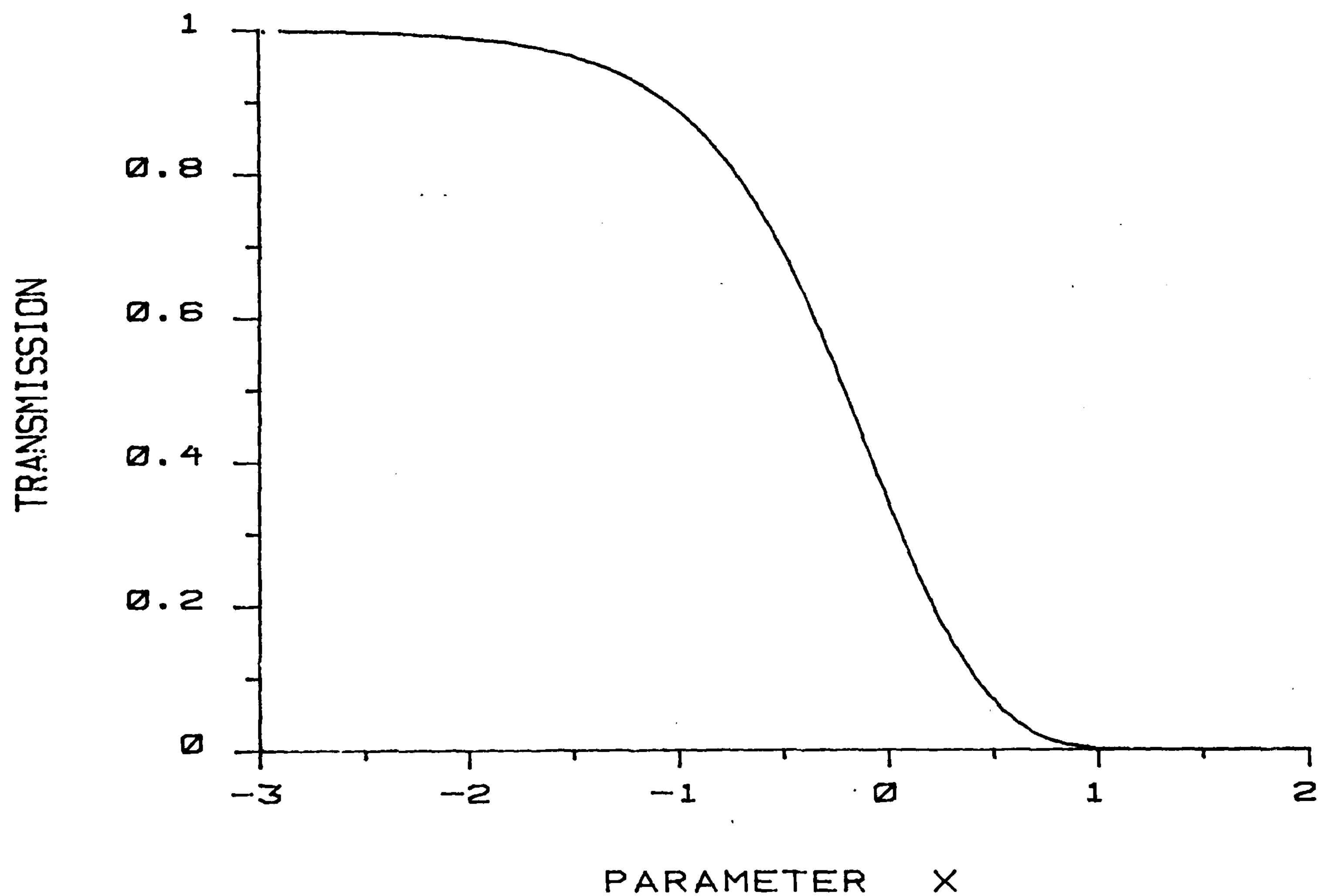


Fig. 9a. Empirical transmission function for SO_2.

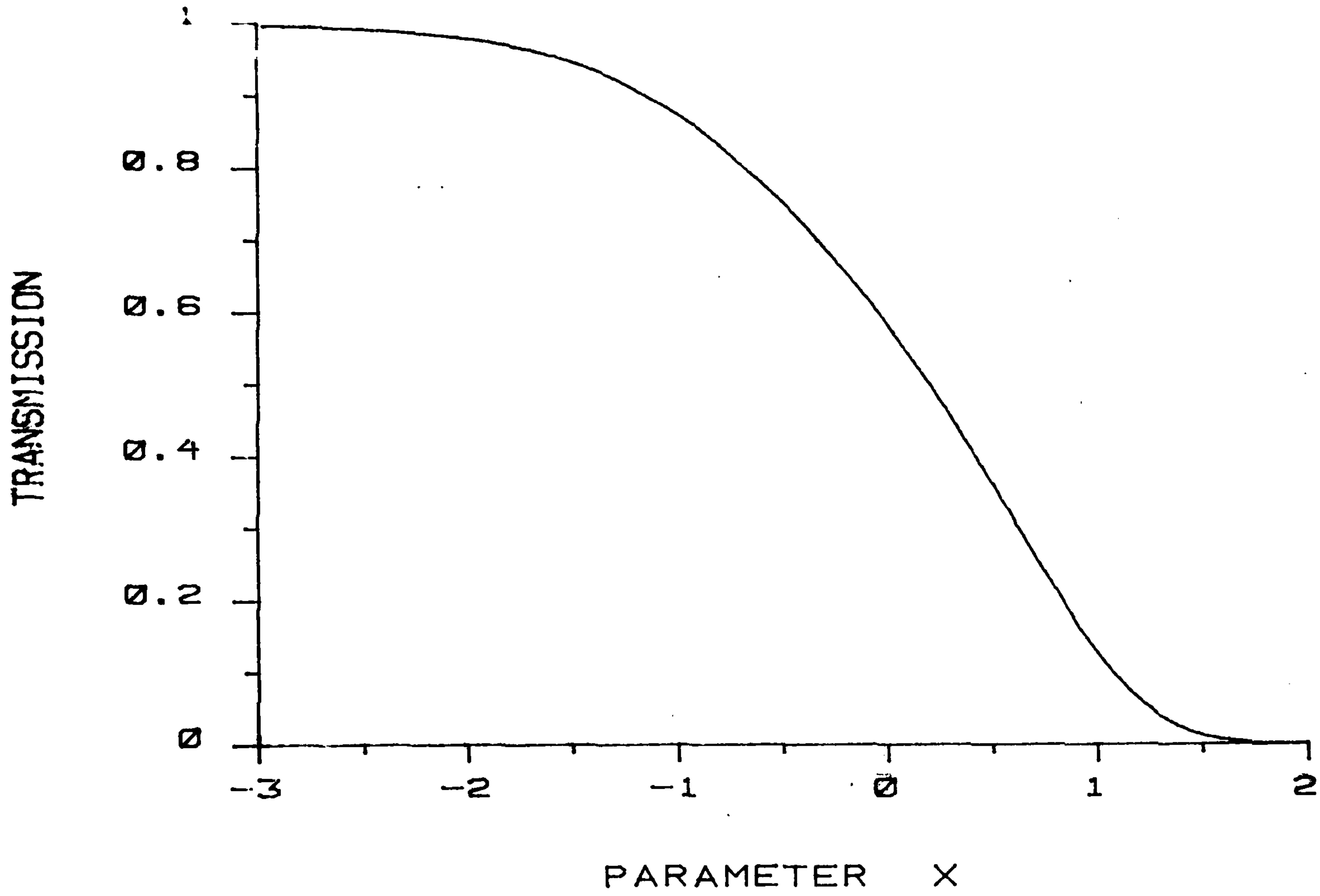


Fig. 9b. Empirical transmission function for NO.

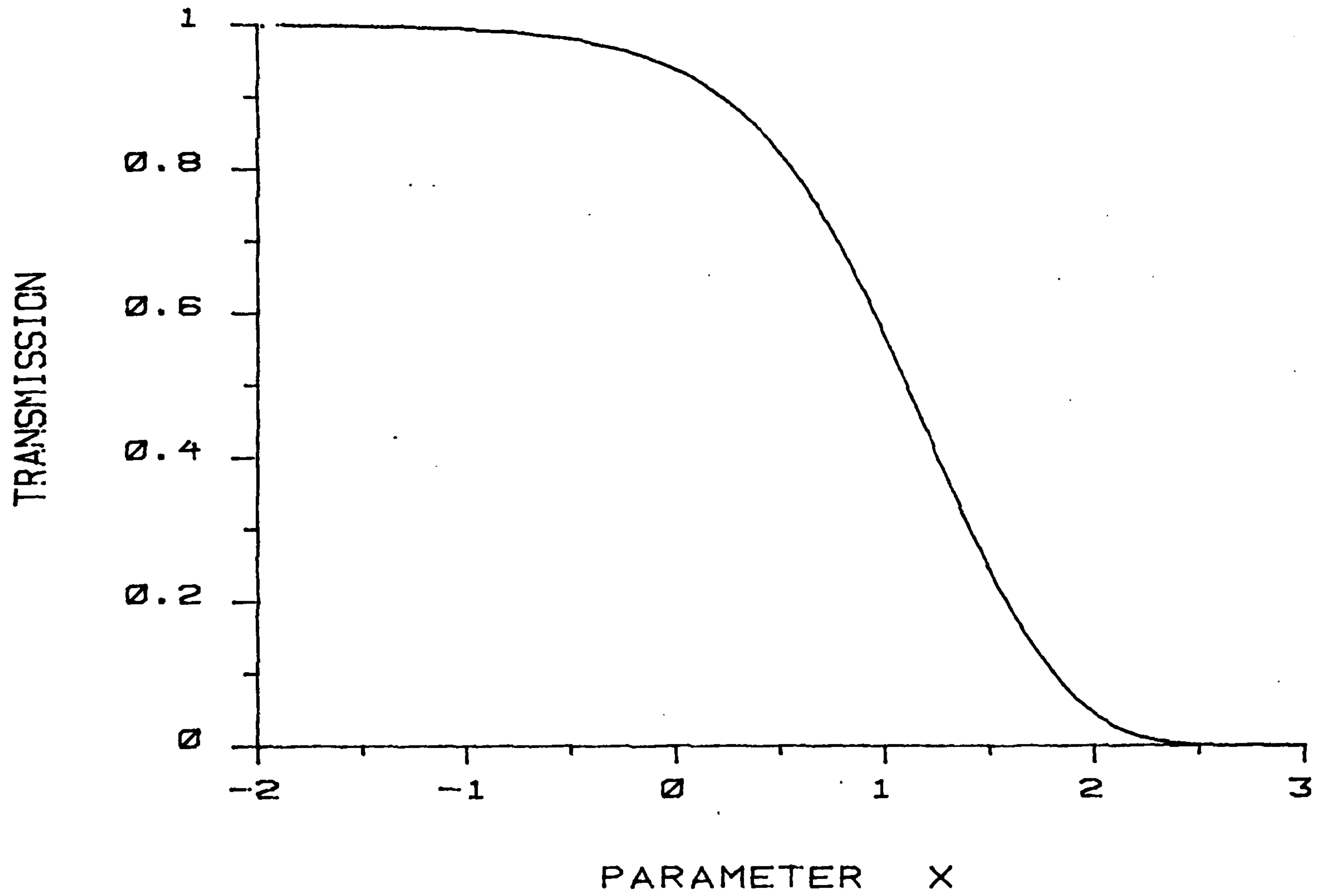


Fig. 9c. Empirical transmission function for NO_2.

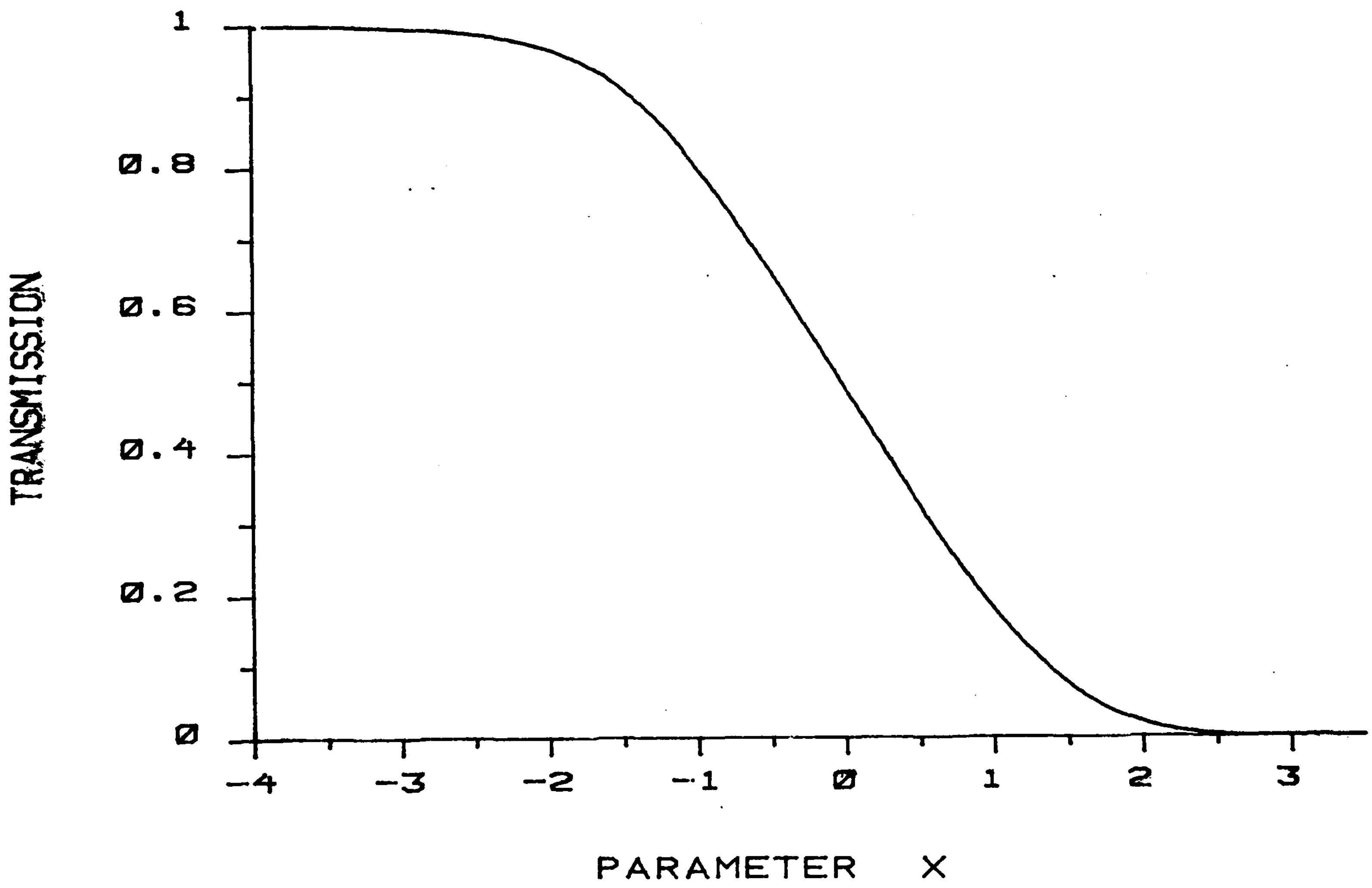


Fig. 9d. Empirical transmission function for NH_3.

VIII. Calculations and Results

8.1 Introduction

The procedure for the use of the Modularized Lowtran in calculations is identical to that of the original and, hence, deserves no further explanation. There are some input and output alterations that deserve some explanatory remarks. Changes in the input format include:

1. Reading of the spectral constants for all band models at the beginning of the main program rather than in the subroutines.
2. Elimination of the transmittance tables for H_2O vapor, infrared O_3 and the uniformly-mixed gases.
3. Reading of the spectral constants for the newly added band models for the trace gases.
4. Reading of the air density profile for the U.S. Standard atmosphere, and of the ppm for the calculation of the equivalent amounts of the trace gases.
5. Changes in the dimension statements to include the additional subscripted variables.

Changes in the output format include:

1. Modification of the print out of the input data.
2. Modification of the output table of computations to include the transmittance for the trace gases.

It should be stressed, however, that the code is operated using exactly the same four control cards as in the original code.

8.2 Testing of Modularized Version

The first step in the testing of the modularization consisted of running identical calculations using the original code and the modularized code before the replacement of the transmittance tables and before the addition of the trace gases. Numerous cases were considered during this effort. A particular case in which the spectral range varied from 2350 to 2450 cm^{-1} for a path at 65° from a height of 2.5 km to a height of 8.5 km and a 23 km visual range, is shown in the Appendix. This output is identical to the output obtained from the original Lowtran.

The second step in the testing procedure consisted of running calculations using the original code and the modularized version with the transmittance tables replaced with the continuous functions, but before the addition of the trace gases. For this purpose, 10 frequencies were selected such that different combinations of models would be effective in the calculation of the total transmittance. The calculations were for a 5 km path at sea level in a sub-arctic winter atmosphere with a 23 km visibility. The results are summarized in Table 9. The columns listed under Transmittance Deviations represent the differences between the calculations using the tabulated and the continuous functions. Note that the average total transmittance deviation is 0.0034, which is below the standard deviation obtained in the curve fitting of

TRANSMITTANCE DEVIATIONS

WAVENUMBER (cm^{-1})	H_2O VAPOR	INFRARED O_3	UNIFORMLY-MIXED GASES	TOTAL TRANSMITTANCE
455	0.0022	0.0000	0.0000	0.0021
555	0.0035	0.0000	0.0026	0.0018
655	0.0041	0.0003	0.0000	0.0000
755	0.0007	0.0003	0.0047	0.0038
955	0.0057	0.0002	0.0050	0.0096
1155	0.0026	0.0003	0.0050	0.0058
1355	0.0044	0.0000	0.0013	0.0006
1855	0.0007	0.0001	0.0034	0.0007
2455	0.0015	0.0000	0.0054	0.0045
3155	0.0037	0.0001	0.0027	0.0053

Table 9. Transmittance difference between calculations using the tabulation of the transmittance functions and calculations using the continuous function representation for a 5 km path at sea level in a sub-arctic winter atmosphere.

the functions to the individual transmittance tables. This deviation amounts to an error of about 0.7% in the middle of the curve-of-growth, which far exceeds the accuracy of Lowtran (between 10 to 20%). The following are attractive features of the continuous functions:

1. They inherently provide for continuous exponential interpolation in transmittance, which is superior to the linear interpolation used in connection with the transmittance tables.

2. They provide for analytical operations such as differentiation and interpolation often needed in radioactive transfer problems.

3. They can be used easily for curve fitting to new transmittance data using computerized procedures.

4. Their use reduces significantly the computer storage requirements for the individual models.

5. They continuously provide for transmittance calculations for small argument values where $0.9999 \leq \tau \leq 1$, for which range Lowtran 4 includes an additional exponential function.

It should be pointed out that the deviations listed in Table 9, although insignificant, do not represent errors solely attributed to the analytical functions. Since they are smaller than the uncertainties in the original data used to develop the tabulated transmittances, they primarily represent differences in the calculational procedures. In fact, in the region between the tabulations the use of the analytical functions are likely to provide more accurate results than the use of the original method in Lowtran.

The last effort in the testing of the modularized code consisted of calculations involving the newly added trace gases. For this purpose ten frequencies were run at which the trace gas models are effective. The same frequencies were run with the modularized Lowtran without these models. The results are summarized in Table 10. The table is primarily intended to show the absorptive effects of the trace gases.

MOLECULAR TRANSMITTANCE

WAVENUMBER (cm^{-1})	SO_2	NO	NO_2	NH_3	TOTAL (with T.G.)	TOTAL (without T.G.)
455	0.9948	1.0000	1.0000	1.0000	0.0000	0.0000
555	0.9308	1.0000	1.0000	1.0000	0.0185	0.0199
655	1.0000	1.0000	0.9997	1.0000	0.0000	0.0000
755	1.0000	1.0000	0.9995	0.9783	0.0029	0.0030
955	1.0000	1.0000	1.0000	0.8878	0.1155	0.1300
1155	0.8929	1.0000	1.0000	0.9820	0.1657	0.1890
1355	0.2728	1.0000	1.0000	1.0000	0.0000	0.0000
1855	1.0000	0.9034	1.0000	1.0000	0.0000	0.0000
2455	0.9998	1.0000	1.0000	1.0000	0.5734	0.5735

Table 10. Calculations of trace gas (T.G.) transmittances for a 5 km path at sea level in a tropical atmosphere with a 23 km visual range. The columns on total transmittance include all the attenuators and the trace gases, except for the rightmost column which excludes the trace gases.

Table 11: (a) Atmospheric regions included in the data calculations

Model	P (mbar)	T (°K)
Standard	1013	288.1
	898.6	281.6
	795.0	275.1
	701.2	268.7
	616.6	262.2
Tropical	805.0	288.0
Subarctic Winter	1013	257.1

(b) Transmittance cuts chosen from the curve of growth

τ_1	0.99
τ_2	0.95
τ_3	0.9
τ_4	0.8
τ_5	0.7
τ_6	0.6
τ_7	0.5
τ_8	0.4
τ_9	0.3
τ_{10}	0.2
τ_{11}	0.1
τ_{12}	0.065

	Absorber Parameters		Spectral Parameter C' (cm^{-1})				Coefficients of Analytical Function	Standard Deviation
	n	m	ν_1 500	ν_2 1165	ν_3 1360	ν_4 2485		
SIMMIN	0.07844	0.06037	0.0	0.019	1.108	-0.566	a_1 = [illegible] a_2 = 0.86759 a_3 = -0.08578	0.006259
Band Model Parameters	0.07130	0.06186	0.0	0.014	1.104	-0.571		

	Empirical										
A	τ_1 =	0.95	0.9	0.8	0.7	0.6	0.5	0.4	0.3	0.2	0.1
	x_1 =	-1.3727	-1.0569	-0.7246	-0.5140	-0.3498	-0.2076	-0.0742	0.0606	0.2115	0.4170

D	Piece-Wise Analytical										Standard Deviation
S	(1st order) a_1 =	0.0682	0.0594	0.0492	0.0408	0.0343	0.0295	0.0273	0.0300	0.0466	
	(1st order) a_2 =	0.9894	0.9811	0.9670	0.9506	0.9319	0.9091	0.8792	0.8353	0.7568	0.005749
E	(1st order) a_3 =	0	0	0	0	0	0	0	0	0	
T	(2nd order) a_1 =	0.0755	0.2247	0.2099	0.1356	0.0590	0.0285	0.0299	0.0151	0.0296	
	(2nd order) a_2 =	1.0016	1.3653	1.5013	1.4061	1.1214	0.8897	0.8715	1.1520	0.8781	0.005604
	(2nd order) a_3 =	0.0050	0.2157	0.4314	0.5273	0.3400	-0.0689	-0.8641	-1.1634	-0.1931	

Table 12a. Band model parameters for SO_2.

	Absorber Parameters		Spectral Parameter C' (cm^{-1})				Coefficients of Analytical Function	Standard Deviation
	n	m	ν_1 1905	ν_2	ν_3	ν_4		
SIMMIN	1.05084	1.08785	0.0				a_1 = −0.26287 a_2 = 0.58035 a_3 = −0.00926	0.008667
Band Model Parameters	0.90099	1.01192	0.0					

A	**Empirical**											
	τ_i =	0.95	0.9	0.8	0.7	0.6	0.5	0.4	0.3	0.2	0.1	
	x_i =	−1.5380	−1.1585	−0.6838	−0.3334	−0.0473	0.1988	0.4193	0.6260	0.8333	1.0715	
D	**Piece-Wise Analytical**											
S	a_1 =	−0.0228	−0.1822	−0.2537	−0.2660	−0.2663	−0.2685	−0.2785	−0.3000	−0.3373		
	(1st order) a_2 =	0.8240	0.6864	0.5818	0.5450	0.5388	0.5497	0.5737	0.6080	0.6528		0.005563
E	a_3 =	0	0	0	0	0	0	0	0	0		
T	a_1 =	−0.1770	−0.1867	−0.2710	−0.2709	−0.2615	−0.3293	−0.4937	−0.6837	−0.2321		
	(2nd order) a_2 =	0.5906	1.0667	0.5046	0.4258	0.6162	1.0007	1.4307	1.6815	0.4283		0.005635
	a_3 =	−0.0866	0.2064	−0.0759	−0.3133	−0.5107	−0.7296	−0.8199	−0.7357	0.1179		

Table 12b. Band model parameters for NO.

	Absorber Parameters		Spectral Parameter C' (cm^{-1})				Coefficients of Analytical Function	Standard Deviation
	n	m	ν_1 775	ν_2 1625	ν_3 2890	ν_4		
SIMMIN	0.19941	0.22631	0.0	2.697	1.271		$a_1 = -1.2203$ $a_2 = 1.0908$ $a_3 = -0.1188$	0.015395
Band Model Parameters	0.17830	0.22484	0.0	2.689	1.259			

DATASET	Empirical											
	τ_1 =	0.95	0.9	0.8	0.7	0.6	0.5	0.4	0.3	0.2	0.1	
	x_1 =	-0.1059	0.2140	0.5543	0.7739	0.9483	0.1028	1.2511	1.4046	1.5784	1.8074	

DATASET	Piece-Wise Analytical										
	(1st order) a_1 =	-1.1865	-1.1824	-1.1656	-1.1404	-1.1052	-1.0604	-1.0042	-0.9381	-0.8656	
	(1st order) a_2 =	0.9770	0.9578	0.9276	0.8950	0.8579	0.8172	0.7723	0.7252	0.6793	0.015555
	(1st order) a_3 =	0	0	0	0	0	0	0	0	0	
	(2nd order) a_1 =	-1.1864	-1.0810	-0.4268	0.0792	0.1053	-0.9289	-3.6587	-5.3854	-0.9659	
	(2nd order) a_2 =	0.9777	0.3013	-1.3600	-1.9671	-1.5164	0.5929	4.7838	6.7090	0.7984	0.015620
	(2nd order) a_3 =	-0.0065	0.8544	1.7222	1.6619	1.1576	0.0953	-1.5105	-2.0059	-0.0352	

Table 12c. Band model parameters for NO_2.

		Absorber Parameters		Spectral Parameter C' (cm^{-1})				Coefficients of Analytical Function	Standard Deviation
		n	m	ν_1	ν_2	ν_3	ν_4		
				930					
	SIMMIN	0.58876	-0.71406	0.0				a_1 = -0.14141 a_2 = 0.44740 a_3 = -0.06716	0.010536
A D S E T	Band Model Parameters	0.52125	-0.60437	0.0					

A	Empirical											
	τ_i =	0.95	0.9	0.8	0.7	0.6	0.5	0.4	0.3	0.2	0.1	
	x_i =	-1.8032	-1.4438	-1.0054	-0.6608	-0.3403	-0.0330	0.2673	0.5751	0.9210	1.3562	
D	Piece-Wise Analytical											
S	(1st order) a_1 =	0.2775	0.0962	-0.0570	-0.1261	-0.1450	-0.1459	-0.1409	-0.1290	-0.1224		
	a_2 =	0.8692	0.7436	0.5913	0.4867	0.4312	0.4037	0.3852	0.3645	0.3573		0.005237
E	a_3 =	0	0	0	0	0	0	0	0	0		
T	(2nd order) a_1 =	0.0894	0.9846	0.2095	-0.1135	-0.1475	-0.1419	-0.2291	-0.3829	-0.0196		
	a_2 =	0.6347	2.2425	1.2594	0.5427	0.3457	0.5098	0.8682	1.0818	0.1698		0.005484
	a_3 =	-0.0722	0.6120	0.4010	0.0559	-0.2290	-0.4530	-0.5734	-0.4794	0.0785		

Table 12d. Band model parameters for NH_3.

AD-A085 436 TEXAS UNIV AT EL PASO DEPT OF ELECTRICAL ENGINEERING F/G 7/4
COMPUTERIZED METHOD FOR THE GENERATION OF MOLECULAR TRANSMITTAN--ETC(U)
APR 80 J H PIERLUISSI, K TOMIYAMA DAAG29-79-C-0067
UNCLASSIFIED ERADCOM/ASL-CR-80-0067-1 NL

2 OF 2
AD
A085436

END
DATE
FILMED
7-80
DTIC

8.3 Band Model Development

Two sets of curves of growth data for each major absorption band for four trace gases SO_2 NO, NO_2, and NH_3 were generated by the line-by-line calculation from the AFGL trace gas parameter tape. One of them consists of 12-cut data for several layers of atmosphere and the other consists of 65-cut data for the standard atmosphere only. Considering the wide range of applications, we included not only the standard atmospheric conditions but also one condition each from the tropical and subarctic winter climates. They are listed in Table 11 together with the 12 chosen transmittance values. The major absorption bands for the four trace gases are given in Table 12 together with the corresponding computed C' values.

Ten middle cuts were chosen from the 12-cut data and used in both ADSET and SIMMIN for the computation of the band model parameters and the standard transmission function. Depending on the number of major absorption bands, the total numbers of data used differ but are in the range of 60-210. The 65-cut data was used in ADSET for the piecewise interpolation to compute piecewise analytical transmission functions.

The ADSET computations were done first. The obtained band model parameter values n, m, and C_1' and nine sets of coefficients a_1, a_2, and a_3 are tabulated in

Table 12 for SO_2, NO, NO_2, and NH_3 in this order. The corresponding standard deviations are alsb listed in these tables.

We also have generated standard atmospheric condition data for non-major bands of each trace gases. These data were used to evaluate non-major $C'(\nu)$ values. The computed $C'(\nu)$ values were listed in Table 5. As we have discussed, these $C'(\nu)$ values and the band model parameters together with the first order piecewise-analytical standard transmission function were implemented in the modularized Lowtran.

We recall that the SIMMIN computation is a recursive one and we need a set of initial guesses of the parameter values to start the computation. For the band model parameters n, m, and C_1', we used the values computed by ADSET. For a_1 and a_2, the respective averages of the first order piecewise interpolation results of ADSET were used. Finally, a_3 was set to be zero. We note that our initial guesses are fairly accurate, since these values were optimal or optimal in average for ADSET computation. A small number ε which was used for the check of convergence was chosen to be 10^{-6}. Since the parameter values are expected to be in the range -10~10, $\varepsilon = 10^{-6}$ gives the limit of numerical accuracy of numbers in the computer. The SIMMIN results are also listed in Table 12.

Typical curve-fits by piecewise analytical standard transmission functions to actual data are shown in Fig. 10 for SO_2 at 500 wavenumber. The corresponding analytical standard transmission function are also compared to the data in Fig. 10. In all of the three graphs in this Figure, the 65-cut data were also plotted to show the fitness of the standard curves.

The computation was repeated using two smaller data sets with 6 and 4 cuts only. The chosen cuts were (0.95, 0.9, 0.8, 0.6, 0.4, and 0.1) for 6 cut data and (0.95, 0.9, 0.6, and 0.2) for 4 cut data. The derived band model parameter values were similar to those in Table 12 and, hence, were not repeated here. Instead, the corresponding standard deviations were listed and compared with the 10 cut cases in Table 13.

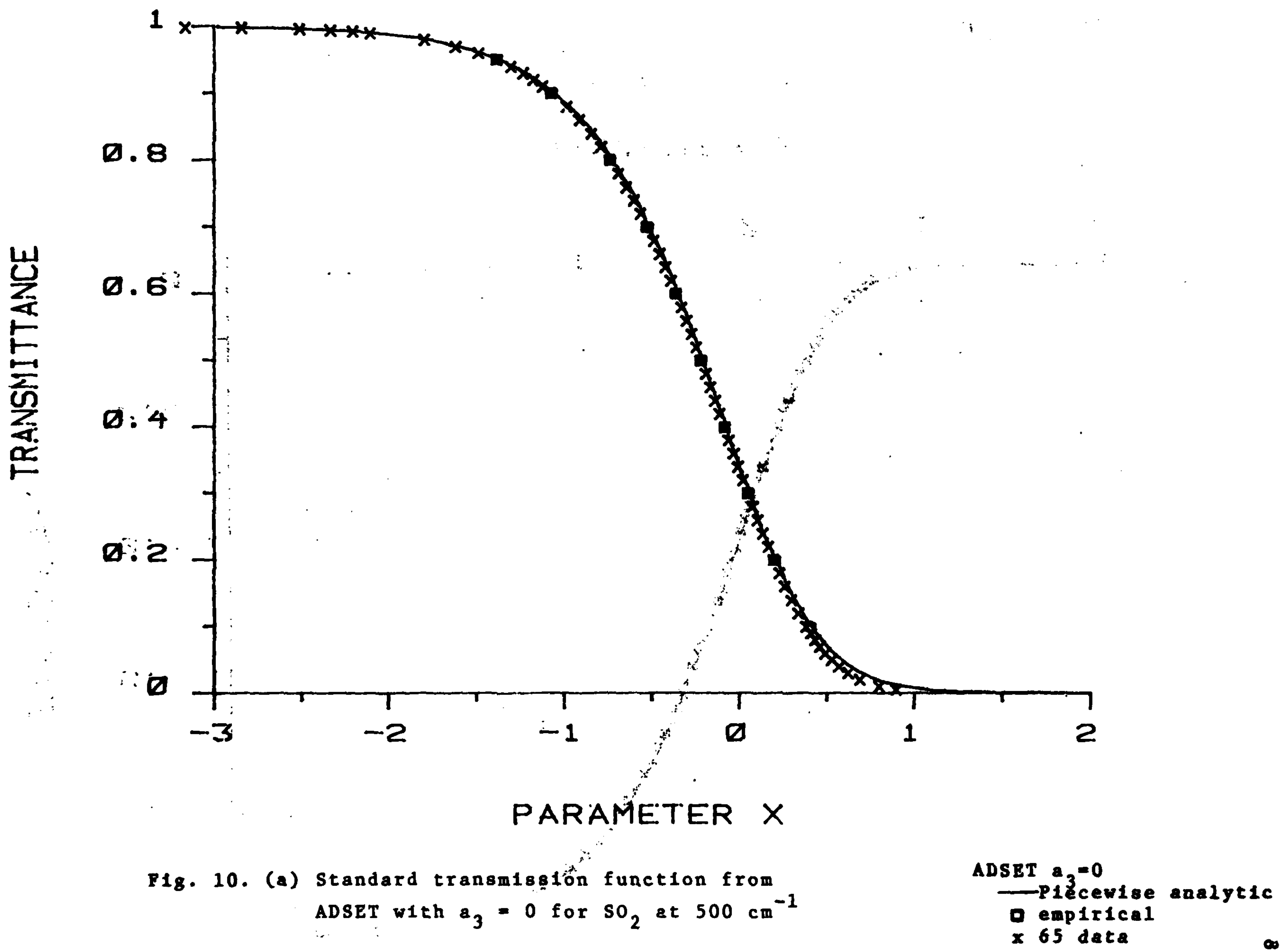


Fig. 10. (a) Standard transmission function from ADSET with $a_3 = 0$ for SO_2 at 500 cm^{-1}

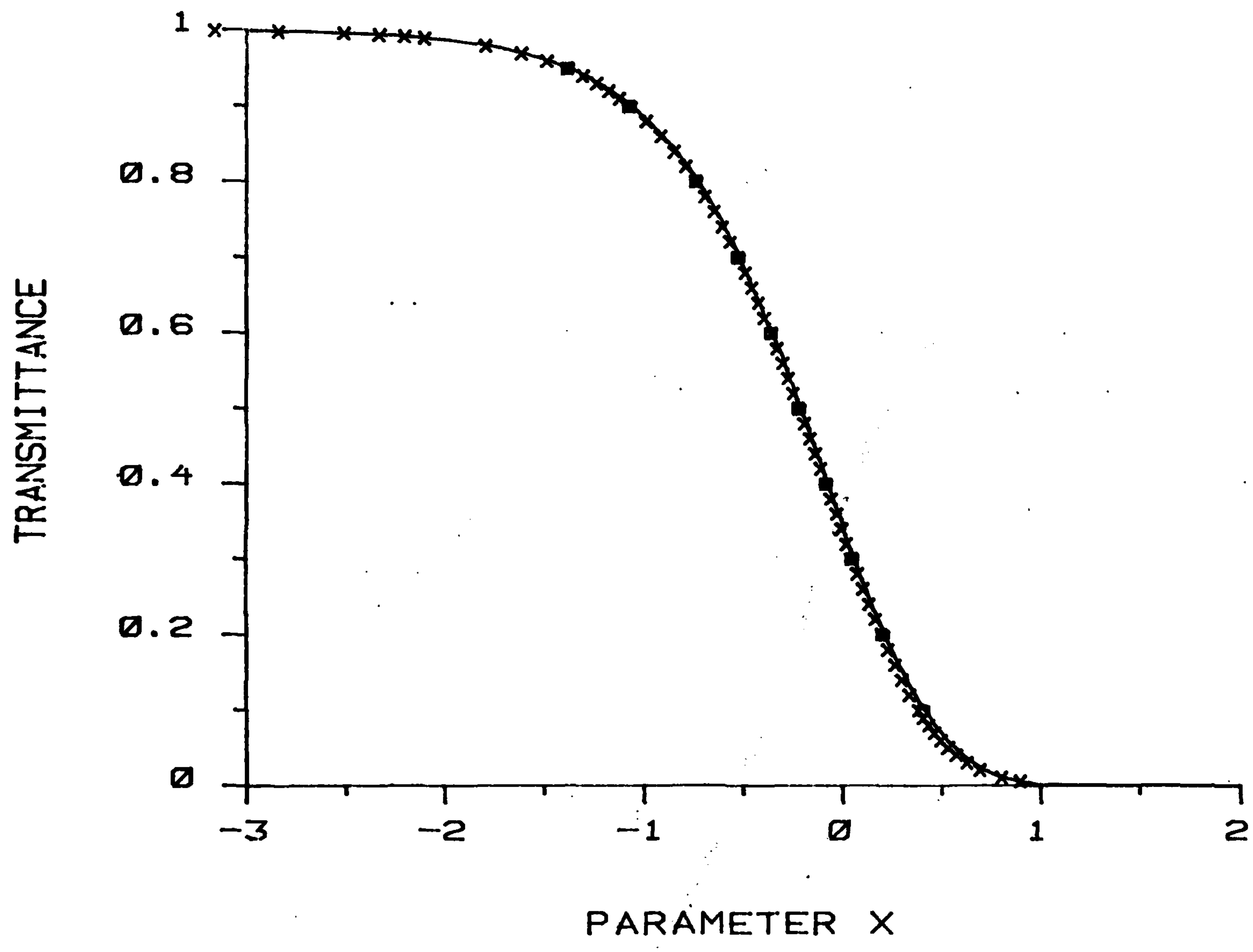


Fig. 10. (b) Standard transmission function from ADSET with $a_3 \neq 0$ for SO_2 at 500 cm^{-1}

ADSET $a_3 \neq 0$
—piecewise analytic
Ø empirical
x 65 data

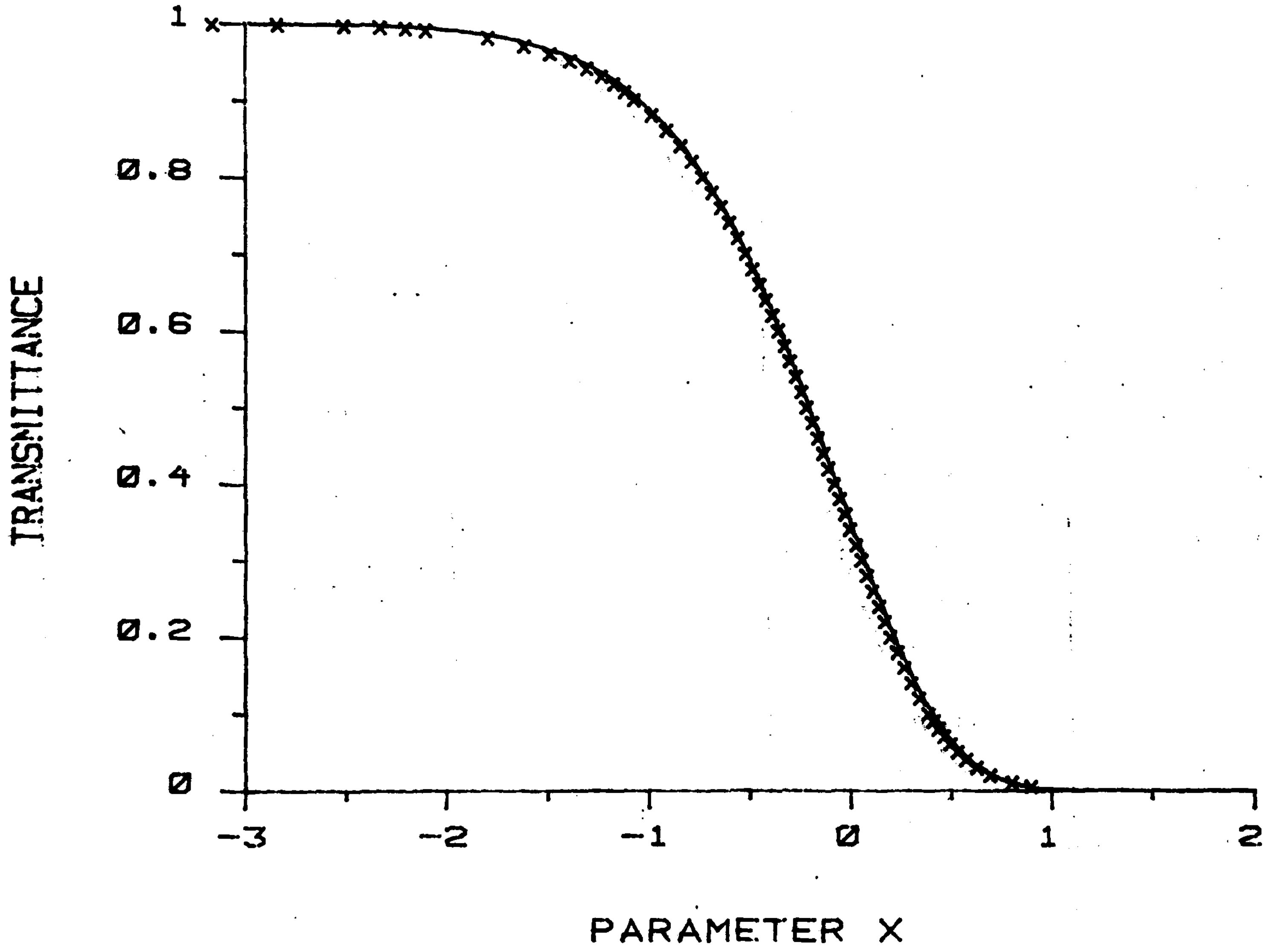


Fig. 10. (c) Standard transmission function from SIMMIN for SO_2 at 500 cm^{-1}

SIMMIN
——: Computed
x : 65 data

ABSORBER	CODES		STANDARD DEVIATIONS IN τ		
			4 Cut Data	6 Cut Data	10 Cut Data
SO_2	SIMMIN		0.004450	0.006636	0.006259
	ADSET	1	0.005344	0.006551	0.005749
		2	0.004830	0.006036	0.005604
NO	SIMMIN		0.005349	0.009345	0.008667
	ADSET	1	0.009310	0.006934	0.005563
		2	0.009210	0.006764	0.005635
NO_2	SIMMIN		0.015009	0.014051	0.015395
	ADSET	1	0.01863	0.015355	0.015555
		2	0.017377	0.014780	0.015620
NH_3	SIMMIN		0.004661	0.010423	0.010558
	ADSET	1	0.006455	0.005454	0.005237
		2	0.006594	0.005400	0.005484

Table 13. Comparison of standard deviations in τ. The two rows, ADSET 1 and 2 are, respectively, for the piecewise analytical transmission functions and linear and quadratic exponents.

IX Discussion and Conclusions

9.1 Introduction

The modularized version presented here is fundamentally the same Lowtran code except for the separation of its computation structure into separate modules or subroutines. Although it is based on the 4th version, it can be adapted with little modification to any future versions, such as the 5th version now in progress. In fact, this latter version already has been structured by AFGL such that the emission/radiance loop is in a subroutine. The modularized code presented here breaks down that loop into a frequency selection subroutine, an equivalent absorber amount subroutine and separate subroutines for each one of the attenuation codes. The use of modules in a complex code such as Lowtran has numerous advantages, among which the amenability for updating by individual users to suit their specific needs is at the top of the list. In the ever changing field of modeling it is highly desirable to be able to easily modify the code for changes in the spectral coverage, the spectral resolution, the absorber concentrations in abnormal environments, the original transmission data used in the development and in the models used for the individual attenuators. The modularized version presented here, although is not the final answer to all conceivable needs, it is a first basic step in that direction. Practicing this predicament, the authors added transmission models for the trace gases to the code.

9.2 Changes

The following are the basic changes introduced in The Modularized Lowtran:

1. The original main program was separated into a central program and subroutines for the absorber amount and the individual attenuation models.

2. In the interest of efficiency and clarity, a new subroutine FGQSL was added for the selection of the attenuation model effective at the given frequency.

3. The subroutine HNO_3 was re-structured to the form of the other previously incorporated subroutines in Lowtran.

4. Continuous analytical models were provided to replace the transmittance curves for H_2O vapor, O_3 and the uniformly-mixed gases.

5. New subroutines for the trace gases SO_2, NO, NO_2 and NH_3 were added.

A copy of the modularized version is found in the Appendix.

9.3 Model Development

The values of band model parameters n and m and spectral parameters C_i' obtained by ADSET and SIMMIN agreed very well. Furthermore, as it was shown in Table 13, the standard deviations corresponding to different cases followed a same pattern for the ADSET and SIMMIN results. This consistency proves the validity of both methods.

In general, the SIMMIN and ADSET computations resulted in similar standard deviations. It was expected that the ADSET computation should result in lower standard deviations since it contained more parameters to adjust. However, for a half of the cases, the SIMMIN code produced lower standard deviations. This is due to the large computational error for the ADSET computations in solving the normal equation $AX = B$. When the condition number of the coefficient matrix A becomes large (i.e., A becomes close to be singular), the computational error becomes so large that it can exceed the directly minimized error of the SIMMIN computation.

We note that this reversal occurred for all four cut data cases. This suggests that the advantage for ADSET of having more parameters to be adjusted is not significant for these cases. Hence, we recommend the use of SIMMIN if the available data contains less than five or six cuts.

A comparison of the standard deviations for two piecewise interpolation results in the ADSET computation showed no significant difference. Furthermore, the results with the second method using quadratic form of x on the exponent of the double exponential function were 'bumpy' for some cases. Since the nature of the transmittance does not predict this behavior, we conclude that the first method using linear function of x is accurate enough to be used in the actual application.

The standard deviations were much higher for NO_2 cases than the cases for the rest of absorbers. By inspecting each curve of growth in detail, it was found that this was mainly due to the difference in the steepness of the curves of growth for three absorption bands. This difference cannot be compensated by C_i' values since they only shift the curves of growth linearly. In fact, within the current band model structure, it is impossible to compensate this difference. Hence, it may be necessary to modify some of the basic assumptions regarding the band model structure, if lower standard deviations are required.

As a side-effect of this discrepancy in the tangent of curves of growth, the SIMMIN computation took far more time for NO_2 cases than the rest. Most of the computations of ADSET were completed by 26-36 CPU seconds. The fluctuations in the computation time were very small. On the other hand, the SIMMIN computation time varied from 14 seconds to 270 seconds. NO_2 cases consumed about 200-270 seconds, which were about four times as much as that for the other cases. This is because the minimizing point in the parameter space is not well defined for NO_2 cases. In other words, the error surface in the parameter space has a very shallow bottom so that the updating step cannot produce large enough changes in the parameter guesses in order to have a rapid convergence.

Thus, it was found that the accuracy of the computed results and the time of execution depend heavily on the actual data. Hence, it is very important to give enough consideration for the data structure. This will be discussed in the next section.

9.4 Data Structure

As it was expressed earlier, we assumed that the number of layers (= the number of data points) in each cut is the same for an absorption band. This was done for the sake of easier coding in data handling. However, this assumption need not be valid. Especially in weaker absorption bands, it is required to use very large range values to have high enough equivalent absorber amounts in order to realize lower transmittances. In some cases the range becomes enormous (in the order of the radius of the earth) so that the corresponding data no longer possess physical significance. The ADSET code has a criterion that if the logarithm of the equivalent absorber, log W, exceeds a certain critical value, then the corresponding data will be set aside and will not be used in the later computation. The critical value was set to be 2 for the actual computation, which corresponds approximately to a vertical path through the atmosphere.

In connection with this, if data are not available at some layers, then the data values are set at 0 to flag the nonavailability of data. ADSET can also detect this and will ignore the data.

A caution must be executed in choosing combinations of pressures and temperatures, i.e., atmospheric conditions. If a data set contains either the standard pressure or the standard temperature or both only, then both ADSET and SIMMIN fail because of the fact that the coefficient of n or m or both in Eq.(12) becomes zero, since

$$\log\left(\frac{P_o}{P_o}\right) = \log 1 = 0, \tag{61}$$

$$\log\left(\frac{T_o}{T_o}\right) = \log 1 = 0. \tag{62}$$

For this case, the coefficient matrix A of the normal equation in ADSET becomes singular and the gradient corresponding to n or m or both in SIMMIN becomes zero all the time. Hence, the normal equation cannot be solved in ADSET and the initial guess of n or m or both cannot be changed in SIMMIN.

Another consideration which should be pointed out is to include different climate conditions. The standard climate condition for several layers of atmosphere contains sequence of pressures and temperatures both of which are monotone decreasing. Therefore, if only these conditions are used, then it is very difficult to distinguish the cause of changes in the transmittance due to the changes in pressure and in temperature. This leads to the shallow bottom of the error surface and hence, large computational error results in ADSET caused by the large condition number of the coefficient matrix and slow convergence in SIMMIN due to the small gradient. In the actual computation, we included not only the standard climate conditions but also one condition each from the tropical and subarctic winter climates in consideration of wide applicability of the results. Numerically speaking, this also resulted in making the regression problem well-posed by breaking the monotonousness of the pressure and temperature combinations of the standard conditions. In fact, several computations were done for ADSET and SIMMIN with standard condition data only. SIMMIN took 10-45 minutes of CPU time to converge if it were convergent and ADSET resulted in a set of absurd values for n and m. Thus, the importance of the numerical consideration, which is ignored in many cases, is clearly indicated. The proper care should be taken when selecting controllable data values.

REFERENCES

1. Elssaser, W.M., "Mean Absorption and Equivalent Absorption Coefficient of a Band Spectrum", Phys. Rev. 34, 126-131 (1938).

2. Pierluissi, J. H., K. Tomiyama and R. B. Gomez, "Analysis of the LOWTRAN Transmission Functions", Appl. Opt. 18, 1607-1612 (1979).

3. Goody, R. M., "A Statistical Model for H_2O Absorption", Quant. J. Roy. Meteorol. Soc., 78, 165, (1952).

4. King, J. I. F., "Modulated Band Absorption Model", J. Opt. Soc. Am., 55, 1498-1503 (1965).

5. Pierluissi, J. H. and K. Tomiyama (1979), "Analytical Determination of the Transmission Functions in the LOWTRAN Code, 1979 IEEE Region V Conference, El Paso, Texas.

6. McClatchey, R. A., et al. (1972), "Optical Properties of the Atmosphere, AFCRL-72-0497, Air Force Cambridge Research Laboratories.

7. Selby, J. E. A. and R. A. McClatchey (1972), Atmospheric Transmittance from 0.25 to 28.5 μm: Computer Code LOWTRAN 2, AFCRL-72-0745, Air Force Cambridge Research Laboratories.

8. ______________________ (1975), Atmospheric Transmittance from 0.25 to 28.5 μm: Computer Code LOWTRAN 3, AFCRL-TR-75-0255, Air Force Cambridge Research Laboratories.

9. Selby, J. E. A., E. P. Shettle and R. A. McClatchey (1976), Atmospheric Transmittance from 0.25 to 28.5 μm: Supplement LOWTRAN 3B, AFGL-TR-76-0258, Air Force Geophysics Laboratory.

10. Selby, J. E. A., et al. (1978), Atmospheric Transmittance/Radiance: Computer Code LOWTRAN 4, AFGL-TR-78-0053, Air Force Geophysics Laboratory.

11. Demshur, P., J. H. Pierluissi and R. B Gomez (1978), Extension of the LOWTRAN Code to Account for Molecular Absorption by the Trace Gases, 1978 Fall AGU Meeting, San Francisco, CA.

12. Pierluissi, J. H., G. A. Gibson and R. B. Gomez (1978), Approximation to the Lorentzian Coefficient for Efficient Calculation of Transmittance Profile, Applied Optics 17, pp. 1425-1431.

13. Pierluissi, J. H., G. Stolt and R. B. Gomez (1979), Comparison of Methods for High-Resolution Transmittance Calculations, 1979 Spring AGU Meeting, Washington, D.C.

14. Pierluissi, J. H., G. A. Gibson and J. Tom Hall (1979), Calculational Error in Laser Molecular Transmittance Due to Uncertainties in the Line and Meteorological Data, SPIE, San Diego, CA.

15. Pierluissi, J. H., "Modularization of the Lowtran Code", Battelle Columbus Lab, Delivery Order 1261, Final Report, August 1979.

16. Gruenzel, R. R., "Mathematical Expressions for Molecular Absorption in LOWTRAN 3B", Appl. Opt., Vol. 17, 2591 (1978).

17. Wismer, D. A. and R. Chattergy, Introduction to Non-linear Optimization, North Holland, New York, Chap. 9, (1978)

18. IBM Manual System/360, Scientific Subroutine Package H20-0205-3, IBM, New York, (1968).

```
FORTRAN IV G LEVEL  21                MAIN              DATE = 79218        16/22/49              PAGE 0001

0001          COMMON /MO1/ Z(34),P(7,34),T(7,34),EH(15,34),WH(7,34),M,NL,FF,CW,C
             1O,PI,WA(7,34)
0002          COMMON /MO2/ IFIND,CA,IP,JSTOP
0003          COMMON /MO3/ HZ1(34),HZ2(6),WO(7,34)
0004          COMMON /MO4/ IHAZE,JP,IM,M1,M2,M3,ML,RO,TBOUND
0005          COMMON /MO5/ C1(2580),C2(1575),C3(540),C4(133),C5(15),C8(102),C11(
             144),C12(115),C13(43),C14(109),C15(95)
0006          COMMON /MO6/ VX(45),C7(45),C7A(45)
0007          COMMON /MO8/ SUM4,SUM5,SUM6,SUM7,SUM8,SUM11
0008          COMMON /MO9/ RADMAX,RADMIN,VRMAX,VRMIN
0009          COMMON /MO10/ FS(9),S1(9),S2(9)
0010          COMMON /MO11/ FNO(9),FN1(9),FN2(9)
0011          COMMON /MO12/ FNH3(9),FH1(9),FH2(9)
0012          COMMON /MO13/ FNO2(9),D1(9),C2(9)
0013          COMMON /MO14/ PPMSO2,PPMNO,PPMNH3,PPMNO2
0014          DIMENSION WPATH(68,15),TBBY(68),WLAY(34,15),HZ(2),TX(15),HMIX(34)
0015          DIMENSION TX9(20),TX12(20),TX13(20),TX14(20),TX15(20),SUMAC(20)
0016          COMMON /EM1/ IEMISS,KMAX,ANGLE,LEN,HMIN,IJ,J1,J2,JMIN,JEXTRA,ITYPE
0017          COMMON /EM2/ IL,IKMAX,LENTOR,NLL,W(15),E(15)
0018          COMMON /EM3/ H1,H2,NP1,MODEL
        C HMIX(I)=HNO3 VOLUME MIXING RATIOS TIMES E+09 FROM EVANS PROFILE
0019          DATA HMIX/9*0.,0.1,0.33,0.8,1.2,1.4,1.6,1.8,1.9,2.0,2.1,2.3,3.0,3.
             17,4.2,5.2,6.0,3.8,2.6,0.22,5*0.0/
        C     PLANCK RADIANCE FUNCTION .........
0020          FF(T,V)=1.190956E-16*(V**5)/(EXP(1.43879*V/T)-1.)
        C       MILLIWATTS. CM-2 ST-1 MICRON-1
0021          F(A)=EXP(18.9766-14.9595*A-2.43882*A*A)*A
0022          DATA HZ(1)/4H23KM/,HZ(2)/4H5 KM/
        C*********************************************************************
        C
        C     MODULARIZED LOWTRAN4: THIS VERSION WAS MODULARIZED BY THE
        C     UNIVERSITY OF TEXAS AT EL PASO FOR THE ATMOSPHERIC SCIENCES LAB.
        C     BY J.H. PIERLUISSI AND S.Y. LIN IN JULY 1979. IT SEPARATES LOWTRAN
        C     INTO MODULES FOR INDIVIDUAL 13 ATTENUATORS, FREQUENCY SELECTION,
        C     AND EQUIVALENT ABSORBER AMOUNT. MODELS FOR THE TRACE GASES SO2,NO,
        C     NO2,NH3 HAVE BEEN ADDED.
        C
        C     PROGRAM MODULARIZED LOWTRAN4 CALCULATES THE TRANSMITTANCE OF THE
        C     ATMOSPHERE FROM 350 CM-1 TO 40000 CM-1 (0.25 TO 28.57 MICRONS) AT
        C     20 CM-1 SPECTRAL RESOLUTION ON A LINEAR WAVENUMBER SCALE.
        C     REFRACTION AND EARTH CURVATURE EFFECTS ARE INCLUDED.    ATMOSPHERE
        C     IS LAYERED IN ONE KM. INTERVALS BETWEEN 0 AND 25 KM., 5 KM. INTER-
        C     VALS TO 50 KM., A TWENTY KM. INTERVAL TO 70 KM., AND A THIRTY KM.
        C     INTERVAL TO 100 KM.
        C*********************************************************************
        C     PROGRAM ACTIVATED BY SUBMISSION OF FOUR CARD SEQUENCE AS FOLLOWS
        C
        C   CARD 1 MODEL,IHAZE,ITYPE,LEN,JP,IM,M1,M2,M3,ML,RO  FORMAT(10I3,F10.3)
        C   CARD 2 H1,H2,ANGLE,RANGE,BETA,VIS                  FORMAT(7F10.3)
        C   CARD 3 V1, V2, DV                                  FORMAT(7F10.3)
        C   CARD 4 IXY                                         FORMAT(I3)
        C
        C     MODEL=1,2,3,4,5 OR 6 SELECTS ONE OF THE FOLLOWING MODEL ATMOSPHERE
        C     TROPICAL,MIDLATITUDE SUMMER,MIDLATITUDE WINTER,SUBARCTIC SUMMER,
        C     SUBARCTIC WINTER,OR THE 1962 U.S.  STANDARD RESPECTIVELY
        C     MODEL=0 FOR HORIZ. PATH WHEN METEOROL. DATA USED:INSTEAD OF CARD 2
        C     READ H1,P(MB),T(DEG C),DEW PT,TEMP(DEG C),REL HUMIDITY,H2O DENSIT
0023          HMIX(29)=1.0E-50
```

```
            C      (GM.M-3),O3 DENSITY(GM.M-3), VIS(KM),RANGE(KM) WITH FORMAT 103 IN
            C      SUBROUTINE ABSORB.
            C      MODEL=7 WHEN NEW MODEL ATMOSPHERE(E.G. RADIOSONDE DATA) USED.
            C      DATA CARDS ARE READ IN BETWEEN CARDS 1 AND 2, AND SHOULD CONTAIN:
            C      ALTITUDE(KM.),PRESSURE,TEMP,DEW PT.TEMP,REL. HUMIDITY,H2O DENSITY,
            C      O3 DENSITY,AEROSOL NO. DENSITY(CM-3) ACCORDING TO FORMAT 103 IN
            C      SUBROUTINE ABSORB.
            C      NOTE THAT EITHER DEW PT. TEMP.OR REL. HUMIDITY CAN BE USED.
            C
            C      M1,M2,M3, ARE USED TO CHANGE TEMP,H2O, AND O3 ALTITUDE PROFILES.
            C         IEMISS=0=TRANSMISSION MODE / IEMISS=1=EMISSION MODE
            C         TBOUND=TEMPERATURE OF EARTH IN DEGREES KELVIN
            C         IF TBOUND=ZERO, ASSUMES AIR TEMPERATURE OF MODEL ATMOS.
            C
            C      IF IHAZE=0 NO AEROSOL SCATTERING IS COMPUTED
            C      IHAZE=1 IF AEROSOL ATTENUATION REQUIRED (THIS IS USED IN
            C      CONJUNCTION WITH VISUAL RANGE (SEE CARD 2))
            C      IHAZE= 1 OR 2 ALSO GIVE AEROSOL ATTENUATION FOR 23KM AND 5KM VIS.
            C      HAZE MODELS RESPECTIVELY IF VIS= 0 ON CARD 2
            C      IHAZE= 7 FOR OTHER AEROSOL MODELS (E.G. MARITIME ECT) WHICH ARE
            C      READ INTO PROGRAM
            C
            C      ITYPE= 1,2 OR 3 INDICATES THE TYPE OF ATMOSPHERIC PATH
            C      ITYPE = 3, VERTICAL OR SLANT PATH TO SPACE
            C      ITYPE = 2, VERTICAL OR SLANT PATH BETWEEN TWO ALTITUDES
            C      ITYPE = 1, CORRESPONDS TO A HORIZONTAL (CONSTANT PRESSURE) PATH
            C
            C      H1 = OBSERVER ALTITUDE (KM)
0024               HMIX(9)=HMIX(29)
            C      H2 = SOURCE ALTITUDE (KM)
            C      ANGLE = ZENITH ANGLE AT H1 (DEGREES)
            C      RANGE = PATH LENGTH (KM)
            C      BETA = EARTH CENTRE ANGLE
            C      VIS = VISUAL RANGE AT SEA LEVEL (KM)
            C      ( IF ITYPE=1 READ H1 AND RANGE; IF ITYPE=3 READ H1 AND ANGLE;
            C      IF ITYPE=2 READ H1 AND TWO OTHER PARAMETERS E.G. H2 AND ANGLE)
            C
            C      V1 = INITIAL FREQUENCY (WAVENUMBER CM-1) INTEGER VALUE
            C      V2 = FINAL FREQUENCY (WAVENUMBER CM-1) INTEGER VALUE
            C      DV = FREQUENCY INTERVALS AT WHICH TRANSMITTANCE IS PRINTED
            C      NOTE: DV MUST BE A MULTIPLE OF 5 CM-1
            C
            C      IXY = 0 TO END DATA, = 1 FOR NEW V1,V2,DV ONLY, = 2 TO CONTINUE
            C      DATA, = 3 FOR NEW CARD 2 ONLY, = 4 FOR NEW CARD 1 ONLY.
            C*********************************************************************
0025               IXY=0
0026               KMAX=15
0027               READ (5,100) IATM,NL
0028               READ (5,101) (HZ1(I),I=1,NL)
0029               READ (5,101) (HZ2(I),I=1,5)
0030               HZ2(6)=HZ1(6)
0031               DO 1 J=1,3
0032               K2=2*J
0033               K1=K2-1
0034               DO 1 I=1,NL
0035        1      READ (5,102) Z(I),(P(K,I),T(K,I),WA(K,I),WH(K,I),WO(K,I),K=K1,K2)
0036               READ (5,132) PPMSO2,PPMNO,PPMNH3,PPMNO2
0037               READ (5,103) (VX(I),C7(I),C7A(I),I=1,44)
```

```
FORTRAN IV G LEVEL  21                  MAIN              DATE = 79218        16/22/49            PAGE
                 READ (5,105) (C1(I),I=1,2580)
                 READ (5,105) (C2(I),I=1,1575)
                 READ (5,105) (C3(I),I=1,540)
                 READ (5,106) (C4(I),I=1,133)
                 READ (5,105) (C5(I),I=1,15)
                 READ (5,106) (C8(I),I=1,102)
                 READ(5,107) (C11(I),I=1,44)
                 READ(5,124) (C12(I),I=1,115)
                 READ(5,125) (F3(I),S1(I),S2(I),I=1,9)
                 READ (5,126) (C13(I),I=1,43)
                 READ (5,126) (C14(I),I=1,139)
                 READ (5,126) (C15(I),I=1,85)
                 READ (5,127) (FN3(I),FN1(I),FN2(I),I=1,9)
                 READ (5,127) (FNH3(I),FH1(I),FH2(I),I=1,9)
                 READ (5,127) (FNO2(I),O1(I),O2(I),I=1,9)
                 C5(1)=0.
                 C5(12)=0.
                 C5(13)=0.
                 C5(14)=0.
                 C5(15)=0.
                 JSTOP=0
                 PI=3.14159265
                 CARDIA=180.
                 IP=0
            2    CONTINUE
                 RE=6371.23
                 IFIND=0
                 READ 100,MODEL,IHAZE,ITYPE,LEN,JP,IM,M1,M2,M3,ML,IEMISS,RO,TBOUND
                 IF(IEMISS.EQ.1) PRINT 108
                 IF(IEMISS.EQ.0) PRINT 109
                 LENTOP=LEN
                 PRINT 100,MODEL,IHAZE,ITYPE,LEN,JP,IM,M1,M2,M3,ML,IEMISS,RO,TBOUND
            3    M=MODEL
                 IF (M.EQ.1) RE=6378.39
                 IF (M.EQ.4) RE=6356.91
                 IF (M.EQ.5) RE=6356.71
                 IF (IHAZE.NE.7) GO TO 4
                 READ 103,(VX(I),C7(I),C7A(I),I=1,44)
                 PRINT 103,(VX(I),C7(I),C7A(I),I=1,44)
                 IHAZE=1
            4    IF (RO.GT.0) RE=RO
            5    CALL ABSORB(IXY,LENTOP,HZ,HMIX,A,TX,V1,V2,DV,ALAM,SUM1,WLAY)
            6    WRITE (6,110)
                 WRITE(6,112) (H(I),I=1,8)
                 WRITE (6,111)
                 WRITE (6,128) (W(I),I=1,15)
                 I=1
                 L=1
                 IV1=V1/5.0
                 IV2=V2/5.+.99
                 IV1=5*IV1
                 IV2=5*IV2
                 IF (IV1.LT.350) IV1=350
                 IF (IV2.GT.50000) IV2=50000
                 IF (DV.LT.5.) DV=5.
                 IDV=DV
                 IV=IV1-IDV
                 NX=0
```

```
      ICOUNT=0
      IF(IEMISS.EQ.0) GO TO 7
      RADSUM=0.0
      FACTOR=0.5
      CALL PATH(WLAY,WPATH,TEBY)
      PRINT 113
      PRINT 114
      IF(IEMISS.EQ.0) IKMAX=IKLO
C**** BEGINING OF TRANSMITTANCE CALCULATIONS
    7 IV=IV+IDV
      NX=NX+1
      SUMV=0.
      TLGLD=1.
      TSOLD=1.
      IKLO=1
      IF(IEMISS.EQ.0) IKLO=IKMAX
      DO 17 IK=IKLO,IKMAX
      IF(IEMISS.EQ.0) GO TO 9
      DO 8 K=1,KMAX
      W(K)=WPATH(IK,K)
    8 CONTINUE
    9 IJ=IK
      IF(IJ.NE.0) GO TO 11
      IF (ICOUNT.EQ.0) GO TO 10
      IF (ICOUNT.EQ.50) GO TO 10
      GO TO 11
   10 ICOUNT=0
      IF(IEMISS.EQ.0) PRINT 115
   11 DO 12 K=1,KMAX
      TX(K)=1.0
   12 CONTINUE
      ICOUNT=ICOUNT+1
      V=IV
      I=(IV-350)/5+1
      SUM4=0.
      SUM5=0.
      SUM6=0.
      SUM7=0.
      SUM8=0.
      SUM11=0.
      CALL FRFOSL(I,IV,W,IHAZE,TX,ALAM)
      TX(9)=SUM4+SUM5+SUM6+SUM7+SUM8+SUM11
      IF (TX(9).EQ.0.0) GO TO 14
      IF(TX(9).LE.0.1) GO TO 13
      IF (TX(9).GT.23.) GO TO 15
      TX(9)=EXP(-TX(9))
      GO TO 16
   13 TX(9)=1.0-TX(9)+0.5*TX(9)*TX(9)
      GO TO 16
   14 TX(9)=1.0
      GO TO 16
   15 TX(9)=0.
   16 TX(9)=TX(1)*TX(2)*TX(3)*TX(9)*TX(12)*TX(13)*TX(14)*TX(15)
      IF (IV.GE.13000) TX(3)=TX(8)
      IF(IEMISS.EQ.0) GO TO 19
      ALAM=1.0E+04/V
      BBIK=FF(TBBY(IK),V)
      TLWFW=(TX(9)*TX(10))/(TX(7)*TX(6))
```

```
          TSNEW=(TX(7)*TX(6))/TX(10)
          DTAU=TLOLD-TLNEW
          IF (DTAU.LT.1.0E-5.AND.TLNEW.LT.1.0E-5) GO TO 18
          SUMV=SUMV+0.5*BBIK*DTAU*(TSOLD+TSNEW)
          TLOLD=TLNEW
          TSOLD=TSNEW
   17     CONTINUE
   18     CONTINUE
          TAUG=0.
          IF(HMIN.LE.0.0.AND.IL.EQ.1) TAUG=TX(9)
          T1=T(M,1)
          IF(TBOUND.GT.0.0) T1=TBOUND
          BBG=FF(T1,V)*TAUG
          IF (HMIN.LE.0) SUMV=SUMV+BBG
          SUMVV=SUMV
          IF (IV.GT.IV1) FACTOR=1.0
          IF (IV.GE.IV2) FACTOR=0.5
          SUMV=(1.0E+04/V**2)*SUMV
          RADSUM=RADSUM+DV*FACTOR*SUMV
          IF(JP.EQ.0) PRINT 116, V,ALAM,SUMV,SUMVV,RADSUM,TX(9)
          IF(SUMV.GE.RADMAX) VRMAX=V
          IF(SUMV.GE.RADMAX)  RADMAX=SUMV
          IF(SUMV.LE.RADMIN) VRMIN=V
          IF(SUMV.LE.RADMIN) RADMIN=SUMV
   19     TX(10)=1.-TX(10)
          AB=1.-TX(9)
          IF(IV.EQ.IV1.OR.IV.EQ.IV2) AB=0.5*AB
          SUMA=SUMA+AB*DV
          IF(IEMISS.EQ.1) GO TO 20
          SUMAC(NX)=SUMA
          TX12(NX)=TX(12)
          TX13(NX)=TX(13)
          TX14(NX)=TX(14)
          TX15(NX)=TX(15)
          TX9(NX)=TX(9)
          IF(JP.EQ.0) WRITE(6,118) IV,ALAM,(TX(K),K=1,7),TX(10)
   20     CONTINUE
          IF (IV.GE.IV2) GO TO 26
          GO TO 7
   26     IV=IV1-IDV
          NX=0
          WRITE(6,129)
   27     IV=IV+IDV
          V=IV
          ALAM=1.0E+04/V
          NX=NX+1
          WRITE(6,130)  IV,ALAM,TX12(NX),TX13(NX),TX14(NX),TX15(NX),SUMAC(NX
         1),TX9(NX)
          IF(IV.GE.IV2) GO TO 21
          GO TO 27
   21     READ 130, IXY
          IF(IEMISS.EQ.1) PRINT 119,VRMIN,RADMIN,VRMAX,RADMAX
          JSTOP=0
          IFIND=0
          AB=1.0-SUMA/(V2-V1)
          PRINT 120, IV1,IV2,SUMA,AB
          IF(IEMISS.EQ.1) PRINT 121,RADSUM
          PRINT 100,IXY
```

```
0210              IF(IXY.EQ.0) GO TO 25
0211              GO TO (22,2,23,24,25),IXY
0212        22    READ 122, V1,V2,DV
0213              AVW=10000./V1
0214              ALAM=10000./V2
0215              PRINT 123, V1,V2,DV,ALAM,AVW
0216              SUMS=0.0
0217              GO TO 6
0218        23    IF(MODEL.EQ.0) GO TO 3
0219              GO TO 5
0220        24    READ 100,MODEL,IHAZE,ITYPE,LEN,JP,IM,M1,M2,M3,ML,IEMISS,RO,TBOUND
0221              IF(IEMISS.EQ.1) PRINT 108
0222              IF(IEMISS.EQ.0) PRINT 109
0223              LENTOK=LEN
0224              PRINT 100,MODEL,IHAZE,ITYPE,LEN,JP,IM,M1,M2,M3,ML,IEMISS,RO,TBOUND
0225              GO TO 3
0226        25    STOP
0227       100    FORMAT(11I3,2F10.3)
0228       101    FORMAT (8F10.3)
0229       102    FORMAT (F6.1,2(F9.3,F5.1,E9.3,2F7.1))
0230       103    FORMAT(4(F6.2,2F7.5))
0231       105    FORMAT (15F5.2)
0232       106    FORMAT (3E9.2)
0233       107    FORMAT(12F6.3)
0234       108    FORMAT('1 PROGRAM WILL BE EXECUTED IN THE EMISSION MODE')
0235       109    FORMAT('1 PROGRAM WILL BE EXECUTED IN THE TRANSMISSION MODE')
0236       110    FORMAT (/10X,38H EQUILAVENT SEA LEVEL ABSORBER AMOUNTS//21X110HWAT
                 1ER VAPOUR      CO2 ETC.        OZONE     NITROGEN (CONT)  H2O (CONT)
                 2    MOL   SCAT       AEROSOL   OZONE(UV-VIS)/24X,7HGM CM-2,10X,2HKM,1
                 30X,6HATM CM,10X,2HKM,9X,7HGM CM-2,10X,2HKM,13X,2HKM,10X,6HATM CM)
0237       111    FORMAT(//21X,64HNITRIC ACID          SO2              NO            NH3
                 1          NO2/)
0238       112    FORMAT (/10X,8H W(1-9)=3(E14.3)/)
0239       113    FORMAT(1H1,50X,'RADIANCE(WATTS/CM2-STER-XXX)')
0240       114    FORMAT (30X,'FR(CM-1)  WVL(MICRON)       PER CM-1    PER MICRON','
                 1  INTEGRAL            TRANS')
0241       115    FORMAT (1H1,/10X,23H  FREQ WAVELENGTH    H2O,5X4HCO2+,5X,52HOZONE
                 1N2 CONT  H2O CONT MOL SCAT  AEROSOL  AEROSOL /11X,14H CM-1  MICRON
                 15,7(4X5HTRANS),4X,5H ABS )
0242       116    FORMAT(30X,F8.1,F13.6,3E13.5,F13.6)
0243       118    FORMAT(10X,I6,9F9.4)
0244       119    FORMAT(' RADMIN ',F12.3,E12.5,/,' RADMAX ',F12.3,E12.5)
0245       120    FORMAT (' INTEGRATED ABSORPTION FROM',I5,' TO',I5,' CM-1 =',F10.2,
                 1',AVERAGE TRANSMITTANCE =',F6.4)
0246       121    FORMAT(' INTEGRATED RADIANCE =',E12.5,'WATT CM -2  SR')
0247       122    FORMAT (7F10.3)
0248       123    FORMAT (/10X,21H FREQUENCY RANGE V1= ,F7.1,13H CM-1 TO V2= ,F7.1,1
                 14H CM-1 FOR DV =,F6.1,9H CM-1   (,F6.2,' - ',F5.2,' MICRONS )')
0249       124    FORMAT(10F8.4)
0250       125    FORMAT(10F8.4)
0251       126    FORMAT(10F8.3)
0252       127    FORMAT(10F8.4)
0253       128    FORMAT(/10X,10H W(11-15)=5(E14.3)/)
0254       129    FORMAT(//10X,23H  FREQ WAVELENGTH    SO2,6X2HNO,7X,31HNH3       NO2
                 1 INTEGRATED  TOTAL/11X,14H CM-1  MICRONS,4(4X5HTRANS),1X,10HABSORP
                 2TION,2X,5HTRANS)
0255       130    FORMAT(10X,I6,7F9.4)
0256       132    FORMAT(4F8.3)
```

```
0257      END
```

```
FORTRAN IV G LEVEL  21              ABSORB             DATE = 79218          16/22/49                PAGE 0001

0001            SUBROUTINE ABSORB(IXY,LENTOR,HZ,HMIX,W,TX,V1,V2,DV,ALAM,SUMA,WLAY)
0002            COMMON /MD1/ Z(34),P(7,34),T(7,34),EH(15,34),WH(7,34),M,NL,RE,CW,C
               1C,P1,WA(7,34)
0003            COMMON /MD2/ IFIND,CA,IP,JSTOP
0004            COMMON /MD3/ HZ1(34),HZ2(6),WC(7,34)
0005            COMMON /MD4/ IHAZE,JP,IM,M1,M2,M3,ML,RC,TBOUND
0006            COMMON /MD9/ RADMAX,RADMIN,VRMAX,VRMIN
0007            COMMON /MD14/ PPMSO2,PPMNO,PPMNH3,PPMNO2
0008            COMMON /EM1/ IEMISS,KMAX,ANGLE,LEN,HMIN,IJ,J1,J2,JMIN,JEXTRA,ITYPE
0009            COMMON /EM3/ H1,H2,RP1,MODEL
0010            DIMENSION AHAZE(34),HSTOR(34),AHZ2(20),WLAY(34,15),VH(15),W(15),E(
               115),TX(15),HMIX(34),HZ(2)
          C***********************************************************************
          C
          C     THIS SUBROUTINE CALCULATES THE EQUIVALENT ABSORBER AMOUNT FOR EACH
          C     GAS AND FOR ANY PATH GEOMETRY
          C
          C     EH(1-15) ARE ABSORBER AMOUNTS PER KM AT LEVEL I, W(1-15) ARE
          C     EQUIVALENT ABSORBER AMOUNTS AT A GIVEN ALTITUDE.
          C
          C***********************************************************************
0011            F(1)=EXP(18.9765-14.9595*A-2.43882*A*A)*A
0012            IF(IXY.EQ.3) GO TO 1
0013            IF(M.EQ.7.AND.IM.NE.0)GO TO 3
0014            IF(IXY.GT.3) GO TO 13
0015            IF (MODEL.EQ.0) GO TO 3
0016      1     READ 100, H1,H2,ANGLE,RANGE,BETA,VIS
0017            PRINT 101, H1,H2,ANGLE,RANGE,BETA,VIS
          C**NOW CONSTRUCT THE PARAMETER FOR CALCULATING CONSTANT PRESSURE PATH QU
0018            X1=RE+H1
0019            IF (ITYPE.EQ.3) GO TO 8
0020            IF (ITYPE.EQ.1) GO TO 13
0021            X2=RE+H2
0022            IF (RANGE.EQ.0.) GO TO 10
0023            PRINT 102, H1,H2,ANGLE,RANGE,BETA,VIS
0024            IF (H2.EQ.0.AND.ANGLE.NE.0) GO TO 2
0025            ANGLE=ARCOS(0.5*((H2-H1)*(1.+X2/X1)/RANGE-RANGE/X1))/CA
0026            GO TO 12
0027      2     X2=SQRT((X1/RANGE+RANGE/X1+2.0*COS(ANGLE*CA))*X1*RANGE)
0028            H2=X2-RE
0029            GO TO 12
0030      3     CONTINUE
0031            IF(ML.LE.0)ML=1
0032            DO 7 K=1,ML
0033            AHAZE(K)=0.0
0034            IF(M.EQ.0)READ 103,H1,P(7,1),TMP,DP,RH,WH(7,K),WO(7,K),VIS,RANGE
0035            IF(M.EQ.0)PRINT 104,H1,P(7,1),TMP,DP,RH,WH(7,K),WO(7,K),VIS,RANGE
0036            IF(M.GT.0)READ 103,Z(K),P(7,K),TMP,DP,RH,WH(7,K),WO(7,K),AHAZE(K)
0037            IF(M.EQ.0)Z(K)=H1
0038            J=IFIX(Z(K)+1.0E-6)+1.
0039            IF(Z(K).GE.25.0) J=(Z(K)-25.0)/5.0+26.
0040            IF(Z(K).GE.50.0) J=(Z(K)-50.0)/20.0+31.
0041            IF(Z(K).GE.70.0) J=(Z(K)-70.0)/30.0+32.
0042            IF(J.GT.33)J=33
0043            FAC=Z(K)-FLOAT(J-1)
0044            IF(J.LT.26) GO TO 4
0045            FAC=(Z(K)-5.0*FLOAT(J-26)-25.)/5.
0046            IF(J.GE.31) FAC=(Z(K)-50.0)/20.
```

```
FORTRAN IV G LEVEL  21                ABSORB            DATE = 79218          16/22/49                PAGE 0002

0047                IF(J.GE.32) FAC=(Z(K)-70.0)/30.
0048                IF(FAC.GT.1.0) FAC=1.0
0049          4     L=J+1
0050                T(7,K)=TMP+273.15
0051                IF(M1.GT.0)T(7,K)=T(M1,J)*(T(M1,L)/T(M1,J))**FAC
0052                TT=273.15/T(7,K)
0053                IF(RH.LE.0.0) TT=273.15/(273.15+DP)
0054                IF(WH(7,K).LE.0.0) WH(7,K)=F(TT)
0055                IF(M2.GT.0)WH(7,K)=WH(M2,J)*(WH(M2,L)/WH(M2,J))**FAC
0056                IF(RH.GT.0.0) WH(7,K)=0.01*RH*WH(7,K)
0057                IF(M3.GT.0)WO(7,K)=WO(M3,J)*(WO(M3,L)/WO(M3,J))**FAC
0058                HSTOR(K)=0.
0059                IF(HMIX(J).LE.0.) GO TO 5
0060                HSTOR(K)=HMIX(J)*(HMIX(L)/HMIX(J))**FAC
0061          5     CONTINUE
0062                IF(Z(K).GE.5.0)GO TO 6
0063                IF(AHAZE(K).EQ.0.0)AHZ2(K)=HZ2(J)*(HZ2(L)/HZ2(J))**FAC
0064          6     IF(AHAZE(K).EQ.0.0)AHAZE(K)=HZ1(J)*(HZ1(L)/HZ1(J))**FAC
0065                IF(MODEL.EQ.0)GO TO 13
0066                IF(K.EQ.1)PRINT 105
0067                PRINT 103,Z(K),P(7,K),TMP,DP,RH,WH(7,K),WO(7,K),AHAZE(K)
0068          7     CONTINUE
0069                IM=0
0070                NL=ML
0071                M1=0
0072                M2=0
0073                M3=0
              C     NOTE THAT Z(I) MAY NOT CORRESPOND TO THE VALUES GIVEN FOR STANDARD
              C     MODEL ATMOSPHERES
0074                IF (IXY.GE.3) GO TO 13
0075                GO TO 1
0076          8     IF (RANGE.GT.0.0) GO TO 9
0077                IF (H2.GT.0.0.AND.H2.LT.H1) IFIND=1
0078                GO TO 13
0079          9     ITYPE=2
0080                BETA=ARCOS(0.5*(RANGE*RANGE/(X1*X2)-X2/X1-X1/X2))/CA
0081         10     IF (BETA.EQ.0.) GO TO 11
0082                IFIND=1
0083                BET=CA*BETA
0084                X2=RE+H2
0085                ANGLE=ATAN(X2*SIN(BET)/(X2*COS(BET)-X1))/CA
0086                RANGE=X2*SIN(BET)/SIN(ANGLE*CA)
0087                BET=BETA
0088                GO TO 13
0089         11     RANGE=(X2/X1)**2-(SIN(ANGLE*CA))**2
0090                IF (RANGE.GE.0.0) RANGE=X1*(SQRT(RANGE)-ABS(COS(ANGLE*CA)))
0091         12     IF (ANGLE.NE.0..OR.ANGLE.NE.180.)BET=ARSIN(RANGE*SIN(ANGLE*CA)/X2)
0092                IF (ANGLE.LT.0.) ANGLE=ANGLE+PI
0093                IF (RANGE.LT.0.0) RANGE=-RANGE
0094                BET=BET/CA
0095                PRINT 102, H1,H2,ANGLE,RANGE,BET,VIS
0096         13     CONTINUE
0097                DO 14 I=1,34
0098                DO 14 J=1,KMAX
0099         14     WLAY(I,J)=0.
0100                SUMA=0.
0101                IF(IXY.LE.2) READ 100,V1,V2,DV
0102                IF(IXY.LE.2)PRINT 100,V1,V2,DV
```

```
      IF (ITYPE.EQ.1) PRINT 106, H1,RANGE
      IF (ITYPE.EQ.2) PRINT 107, H1,H2,ANGLE
      IF (ITYPE.EQ.3) PRINT 108, H1,ANGLE
      IF (MODEL.EQ.0) M=7
      IF(VIS.GT.0.0) PRINT 109,VIS
      IF(VIS.LT.2.0.AND.VIS.GT.0.0) PRINT 110
      IF (M.EQ.1) PRINT 111, M
      IF (M.EQ.2) PRINT 112, M
      IF (M.EQ.3) PRINT 113, M
      IF (M.EQ.4) PRINT 114, M
      IF (M.EQ.5) PRINT 115, M
      IF (M.EQ.6) PRINT 116, M
      IF (IHAZE.EQ.0.) PRINT 117
      IF(VIS.LE.0.0.AND.IHAZE.GT.0) PRINT 118,IHAZE,HZ(IHAZE)
      AVW=10000./V1
      ALAM=10000./V2
      RADMIN=1.0E+74
      RADMAX=0.
      VRMIN=0.
      VRMAX=0.
      PRINT 119, V1,V2,DV,ALAM,AVW
      AVW=0.5E-4*(V1+V2)
      AVW=AVW*AVW
      CD=77.46+.459*AVW
      CW=43.487-0.3473*AVW
      IF (IFIND.EQ.1) GO TO 19
   15 IF (IFIND.EQ.1) CALL ANGL (H1,H2,ANGLE,BETA,LEN,NL)
      IFIND=0
      IF (JP.EQ.0) PRINT 120
      IF (ITYPE.EQ.1) GO TO 19
      DO 16 K=1,KMAX
      VH(K)=0.0
   16 CONTINUE
      BETA=0.0
      SR=0.0
      IP=0
C**** NOW DEFINE CONSTANT PRESSURE PATH QUANTITIES  EH(I-8)
      Y=CA*ANGLE
      SPHI=SIN(Y)
      F1=(RE+H1)*SPHI
      IF (H1.GT.Z(NL)) GO TO 17
      GO TO 19
   17 X=(RE+Z(NL))/(RE+H1)
      IF (SPHI.GT.X) GO TO 18
      H1=Z(NL)
      J1=NL
      SPHI=SPHI/X
      ANGLE=180.0-ARSIN(SPHI)/CA
      F1=(RE+H1)*SPHI
      GO TO 19
   18 HMIN=F1-RE
      PRINT 122, HMIN
      GO TO 59
   19 DO 22 I=1,NL
      PS=P(M,I)/1013.0
      TS=273.15/T(M,I)
      IF(M1.GT.0.AND.M.LT.7)TS=273.15/T(M1,I)
      X=PS*TS
```

```
FORTRAN IV G LEVEL  21            ABSORB            DATE = 79218        16/22/49            PAGE 0004

0160              PT=PS*SQRT(TS)
0161              D=0.1*WH(M,I)
0162              IF(M2.GT.0.AND.M.LT.7) D=0.1*WH(M2,I)
0163              EH(1,I)=D*PT**0.9
0164              EH(2,I)=X*PT**0.75
0165              EH(4,I)=0.8*PT*X
0166              PPW=4.56E-5*D*273.15/TS
0167              TS1=(296.0/273.15)*TS
0168              EH(5,I)=D*PPW*EXP(6.08*(TS1-1.0))+0.002*D*(PS-PPW)
0169              EH(10,I)=D*(PPW+0.12*(PS-PPW))*EXP(4.56*(TS1-1.0))
0170              EH(6,I)=X
0171              HAZE=HZ1(I)
0172              IF(M.EQ.7) HAZE=AHAZE(I)
0173              IF(Z(I).GE.5.0) GO TO 20
0174               IF(M.NE.7.AND.IHAZE.EQ.2) HAZE=HZ2(I)
0175              IF(IHAZE.EQ.2.AND.M.EQ.7)HAZE=AHZ2(I)
0176              IF(VIS.LE.0.0) GO TO 20
0177              IF(M.NE.7)HAZE= 6.389*((HZ2(I)-HZ1(I))/VIS+HZ1(I)/5.0-HZ2(I)/23.0)
0178              IF (M.NE.7) GO TO 20
0179              HAZE=6.389*((AHZ2(I)-AHAZE(I))/VIS+AHAZE(I)/5.0-AHZ2(I)/23.0)
0180       20     IF(HAZE.LT.0.0) HAZE=0.0
0181              EH(7,I)=HAZE/HZ1(I)
0182              IF(MODEL.EQ.7) EH(7,I)=HAZE/AHAZE(I)
0183              EH(8,I)=46.6667*WO(M,I)
0184              IF(M3.GT.0.AND.M.LT.7) EH(8,I)=46.667*WO(M3,I)
0185              EH(3,I)=EH(8,I)*PT**0.4
            C EH(11,I)=HNO3 ABSORBER AMOUNT (ATM-CM)/KM
0186              EH(11,I)=PS*TS*HMIX(I)*1.0E-04
0187              IF (MODEL.EQ.0.OR.MODEL.EQ.7) EH(11,I)=PS*TS*HSTOR(I)*1.0 E-04
            C*****EH(12,I) = SO2 ABSORBER AMOUNT (ATM-CM)/KM
0188              EH(12,I)=0.772E-04*PPMSO2*WA(M,I)*PS**0.07122*TS**0.06159
0189              EH(13,I)=0.772E-04*PPMNO*WA(M,I)*PS**0.90098*TS**1.01192
0190              EH(14,I)=0.772E-04*PPMNH3*WA(M,I)*PS**0.52125*TS**(-0.60438)
0191              EH(15,I)=0.772E-04*PPMNO2*WA(M,I)*PS**0.18066*TS**0.20911
0192              EH(9,I)=1.0
0193              REF=1.0E-6*(CO*X*1013.0/273.15-PPW*CW)
0194              IF (I.EQ.NL) GO TO 21
0195              IF (MODEL.EQ.0.AND.I.GE.1) GO TO 30
0196              T2=T(M,I+1)
0197              W2=WH(4,I+1)
0198              IF(M1.GT.0)T2=T(M1,I+1)
0199              IF(M2.GT.0)W2=WH(M2,I+1)
0200              PPW=4.56E-6*W2*T2
0201              EH(9,I)=0.5*(REF+1.0E-6*(CO*P(M,I+1)/T2-PPW*CW))
0202       21     IF (I.EQ.NL) EH(9,I)=0.
0203              IF (H1.GE.Z(I)) J1=I
0204              IF(IFIND.EQ.0.OR.JP.EQ.0) PRINT 123, I,Z(I),(EH(K,I),K=1,12)
0205              EH(9,I)=EH(9,I)+1.0
0206       22     CONTINUE
0207       23     IF (IFIND.EQ.1) GO TO 15
0208              IP=1
0209              IK=0
0210              X1=H1
0211              CALL POINT (H1,YN,N,NP1,TX,IP)
0212              J1=N
0213              TX1=TX(9)
0214              DO 24 K=1,KMAX
0215       24     E(K)=TX(K)
```

```
C**ITYPE=1 MEANS HORIZONTAL PATH ******************************************
      IF (ITYPE.EQ.1) GO TO 35
      IF (ITYPE.EQ.3) H2=Z(NL)
C** ANGLE GREATER THAN 90 DEGREES MEANS DOWNWARD TRAJECTORY ************
      IF (ANGLE.GT.90.0) GO TO 38
C** IF THE PATH IS NOT HORIZONTAL OR DOWNWARD THEN IT IS UPWARD TRAJECTORY
   25 IF (ANGLE.GT.90.0.AND.NP1.GT.0) J1=J1+1
      J2=NL
      IF (ITYPE.EQ.3) GO TO 26
      CALL POINT (H2,YN,N,NP,TX,IP)
      J2=N
      IF (NP.GT.0) J2=J2-1
   26 DO 27 K=1,KMAX
      IF(K.EQ.9) GO TO 27
      FH(K,J1)=E(K)
      IF (ITYPE.EQ.3) GO TO 27
      FH(K,J2+1)=TX(K)
   27 CONTINUE
      IF (J1.EQ.J2) TX1=TX1+YN-FH(9,J1)
C**** NOW DEFINE VERTICAL PATH QUANTITIES VH(1-8)
      IF (JP.EQ.0) PRINT 124
      DO 28 K=1,KMAX
   28 W(K)=0.
      DO 35 I=J1,J2
      X1=Z(I)
      X2=Z(I+1)
      IF (I.EQ.J1) X1=H1
      IF (I.EQ.J2) X2=H2
      DZ=X2-X1
      IF (I.EQ.NL) DZ=Z(I)-Z(I-1)
      DS=DZ
C***** UPWARD TRAJECTORY
      RX=(RE+X1)/(RE+X2)
      THETA=ARSIN(SPHI)/CA
      PHI=ARSIN(SPHI*RX)/CA
      BET=THETA-PHI
      SALP=RX*SPHI
      IF (SPHI.GT.1.E-10) DS=(RE+X2)*SIN(BET*CA)/SPHI
      BETA=BETA+BET
      PSI=BETA+PHI-ANGLE
      PHI=180.-PHI
      SR=SR+DS
      JEXTRA=0
      DO 34 K=1,KMAX
      EV=DS*FH(K,I)
      IF (I.EQ.NL) GO TO 29
      IF (FH(K,I).EQ.0.0.OR.FH(K,I+1).EQ.0.0) GO TO 30
      IF (FH(K,I).EQ.FH(K,I+1)) GO TO 31
      EV=DS*(FH(K,I)-FH(K,I+1))/ALOG(FH(K,I)/FH(K,I+1))
      GO TO 31
   29 IF (FH(K,I).EQ.0.0) GO TO 30
      IF (FH(K,I-1).EQ.0.0) GO TO 30
      IF (FH(K,I).EQ.FH(K,I-1)) GO TO 31
      EV=EV/ALOG(FH(K,I-1)/FH(K,I))
      GO TO 31
   30 EV=0.
   31 VH(K)=VH(K)+EV
      IF(I.EQ.JSTOR) GO TO 33
```

```
FORTRAN IV G LEVEL  21                ABSORB            DATE = 79218          16/22/49              PAGE 0006

 0269         32    WLAY(I,K)=EV+W(K)
 0270               W(K)=0.
 0271               GO TO 34
 0272         33    W(K)=EV
 0273               IF(J1.NE.J2) GO TO 34
 0274               WLAY(J2+1,K)=W(K)
 0275               W(K)=0.
 0276               JEXTRA=1
 0277         34    CONTINUE
 0278               IF (JP.EQ.0) PRINT 125, I,K1,(VH(L),L=1,8),PSI,PHI,BETA,THETA,SR
 0279               IF (I.GE.NL) GO TO 35
 0280               IF (I+1.EQ.J2) EH(9,I+1)=YN
 0281               IF (I.EQ.J1) EH(9,I)=TX1
 0282               RN=EH(9,I+1)/EH(9,I)
 0283               SPHI=SPHI*RX/RN
 0284               IF (SALP.GE.RN) SPHI=SALP
 0285         35    CONTINUE
 0286               GO TO 58
              C**** HORIZONTAL PATH
 0287         36    DO 37 K=1,KMAX
 0288               W(K)=RANGE*EH(K,I)
 0289               IF (MODEL.GT.0) W(K)=RANGE*TX(K)
 0290               VH(K)=W(K)
 0291         37    CONTINUE
 0292               GO TO 60
 0293         38    CONTINUE
              C**** DOWNWARD TRAJECTORY
 0294               K2=0
 0295               IF (NPI.EQ.1) J1=J1-1
 0296               J2=J1+1
 0297               J=J1+1
 0298               YN1=YN
 0299               IF (H2.GT.Z(J1+1).OR.H1.EQ.H2) GO TO 40
 0300               IF (NP1.EQ.1.AND.H2.GE.Z(J1+1)) GO TO 40
 0301               CALL POINT (H2,YN,N,NP2,TX,IP)
 0302               DO 39 K=1,KMAX
 0303         39    W(K)=TX(K)
 0304               TX2=TX(9)
 0305               YN2=YN
 0306               IF (H2.LT.H1) H=H2
 0307               J2=N
 0308               IF (J1.EQ.J2) TX2=TX1+YN2=EH(9,N)
 0309               IF (H2.GT.H1) TX1=TX2
 0310               IF (J1.EQ.J2.AND.H2.LT.H1) YN1=TX2
 0311         40    AO=(RE+H1)*SPHI*YN1
 0312               IF (H2.GE.H1) YN2=YN1
 0313               DO 41 I=1,J1
 0314               HMIN=AO/EH(9,I)-RE
 0315               IF (I.EQ.J1) HMIN=AO/YN1-RE
 0316               JMIN=I
 0317               IF (HMIN.LE.Z(I+1)) GO TO 42
 0318         41    CONTINUE
 0319         42    X=HMIN
 0320               IF (HMIN.LE.0) GO TO 44
 0321               CALL POINT (X,YN,N,NP,TX,IP)
 0322               JMIN=N
 0323               TX3=TX(9)
 0324               IF (J2.EQ.N.OR.J1.EQ.N) TX3=YN2+TX(9)=EH(9,N)
```

```
      IF(TX3.LT.0.0)TX3=TX(9)
      IF (J1.EQ.N.AND.H2.GE.H1) GO TO 43
      HMIN=A3/TX3-RE
      IF (ABS(X-HMIN).GT.0.0001) GO TO 42
   43 IF (J1.EQ.N.AND.H2.GE.H1) YN1=TX3
      IF (J2.EQ.N.AND.J1.NE.J2) YN2=TX3
      IF (H2.GE.H1) TX2=TX3
      IF (H2.GE.H1) J2=N
      IF (H2.GE.H1.OR.H2.LT.HMIN) H=HMIN
      PRINT 126, HMIN
      IF(H2.LT.HMIN)J2=N
      IF (H2.LT.HMIN) PRINT 127, HMIN
      GO TO 45
   44 PRINT 126,HMIN
      IF (H2.LT.H1) GO TO 45
      IF (ITYPE.EQ.3.OR.H2.GE.H1) PRINT 128
      ITYPE=2
      TX2=EH(9,1)
      JMIN=0
      J2=1
      H2=0.0
      H=0.0
C**** NOW DEFINE VERTICAL PATH QUANTITIES VH(1-8)
   45 IF(JP.EQ.0) PRINT 124
      JSTOR=J-1
      DO 51 I=1,NL
      J=J-1
      FF=EH(9,J)
      IF (I.EQ.1) REF=YN1
      IF (I.EQ.1.AND.K2.EQ.1) REF=YN2
      IF (J.EQ.J2.AND.K2.EQ.0) REF=TX2
      IF (I.NE.1) X1=Z(J+1)
      X2=Z(J)
      IF (J.EQ.J2.AND.K2.EQ.0) X2=H
      IF (J.EQ.JMIN.AND.K2.EQ.1) X2=HMIN
      HM=(RF+X1)*SPHI-RE
      IF (HM.GT.Z(J).AND.HM.GT.X2) X2=HM
      RX=(RE+X1)/(RE+X2)
      DS=X1-X2
      ALP=90.0
      THET=ARSIN(SPHI)/CA
      SALP=RX*SPHI
      IF (ABS(X2-HM).GT.1.0E-5) ALP=ARSIN(SALP)/CA
      BET=ALP-THET
      IF (SPHI.GT.1.0E-10) DS=(RE+X2)*SIN(BET*CA)/SPHI
      THETA=180.0-THET
      BETA=BETA+BET
      PSI=BETA-ALP-ANGLE+180.0
      SP=SR+DS
      DO 50 K=1,KMAX
      AJ=EH(K,J)
      BJ=EH(K,J+1)
      IF (J.EQ.J1) BJ=E(K)
      IF (J.EQ.J2.AND.H2.LT.H1.AND.H2.GT.0.0) AJ=W(K)
      IF (J.EQ.JMIN.AND.H2.GE.H1) AJ=TX(K)
      IF (J.EQ.JMIN.AND.ABS(H2-HM).LT.1.0E-5) AJ=TX(K)
      IF (K2.EQ.0) GO TO 46
      IF (J.EQ.J2) BJ=W(K)
```

```
0382              IF (J.EQ.JMIN) AJ=TX(K)
0383        46    IF (AJ.EQ.0.0.OR.BJ.EQ.0.0) GO TO 48
0384              IF (AJ.EQ.BJ) GO TO 47
0385              EV=DS*(AJ-BJ)/ALOG(AJ/BJ)
0386              GO TO 49
0387        47    EV=DS*BJ
0388              GO TO 49
0389        48    EV=0.0
0390        49    VH(K)=VH(K)+EV
0391        50    WLAY(J,K)=EV
0392              IF (JP.EQ.0) PRINT 125, J,X1,(VH(L),L=1,8),PSI,ALP,BETA,THETA,SR
0393              IF (J.EQ.J2.AND.H2.GE.H1) GO TO 56
0394              IF (J.EQ.JMIN.AND.K2.EQ.1) GO TO 54
0395              IF (J.NE.1) RN=REF/EH(9,J-1)
0396              IF (J.EQ.J2+1) RN=REF/TX2
0397              IF (J.EQ.J2.AND.K2.EQ.0) RN=REF/YN2
0398              IF (J.EQ.(JMIN+1).AND.K2.EQ.1) RN=REF/TX3
0399              IF (SALP.GE.RN) RN=1.0
0400              SPHI=SALP*RN
0401              IF (J.EQ.J2.AND.K2.EQ.0) GO TO 52
0402        51    CONTINUE
0403        52    IF (HMIN.LE.0) GO TO 58
0404              IF (LEN.EQ.0) PRINT 129
0405              IF (LEN.EQ.0) GO TO 58
0406              IF (LEN.EQ.1) PRINT 130
0407              K2=1
0408              X1=X2
0409              IF (ABS(X1-HMIN).LE.0.001) GO TO 58
0410              H=HMIN
0411              J=J2+1
0412              IF (NP2.EQ.1) J=J-1
0413              B=BETA
0414              PH=180.0-ARSIN(SPHI)/CA
0415              TS=SR
0416              PS=PSI
0417              DO 53 K=1,KMAX
0418        53    E(K)=VH(K)
0419              GO TO 45
0420        54    BETA=2.*BETA-B
0421              PSI=2.*PSI-PS
0422              SR=2.*SR-TS
            C      LONG PATH  TAKEN
0423              PHI=PH
0424              DO 55 K=1,KMAX
0425        55    VH(K)=2.*VH(K)-E(K)
0426              GO TO 58
0427        56    DO 57 K=1,KMAX
0428        57    VH(K)=2.0*VH(K)
0429              BETA=2.0*BETA
0430              SR=2.0*SR
0431              IF (H2.EQ.H1) GO TO 58
0432              RN=TX1/YN1
0433              SPHI=SIN(ANGLE*CA)
0434              IF (SPHI.LT.RN) SPHI=SPHI/RN
0435              GO TO 25
0436        58    CONTINUE
0437              IF(ANGLE.GT.90.0) PRINT 100,HP
0438              DO 59 K=1,KMAX
```

```
FORTRAN IV G LEVEL  21                ABSORB            DATE = 79218         16/22/49                PAGE 0009

0439              W(K)=VH(K)
0440        59    CONTINUE
0441        60    RETURN
0442       100    FORMAT (7F10.3)
0443       101    FORMAT (10X,7F10.3)
0444       102    FORMAT (10X,' H1=',F7.3,'KM,H2=',F7.3,'KM,ANGLE=',F8.4,'GEOM. RANG
                 1E =',F7.2,'KM,BETA=',F8.5,',VIS=',F6.1)
0445       103    FORMAT(3F10.3,2F5.1,2E10.3,2F10.3)
0446       104    FORMAT(10X,'INPUT METEOROLOGICAL DATA:'/10X,'Z=',F7.2,' KM, P=',F7
                 1.2,' MB,T=',F5.1,' C, DEW PT.TEMP',F5.1,' C, REL HUMIDITY=',F5.1,
                 2' %, H2O DENSITY=',1PE9.2,' GM M-3'/10X,' OZONE DENSITY=',E9.2,' G
                 3M-3, VISUAL RANGE=',0PF6.1,' KM,RANGE=',F10.3,' KM ' )
0447       105    FORMAT('  MODEL ATMOSPHERE NO. 7',/ 4X,'Z (KM)',3X,'P (MB)',4X,
                 1 'T (C) DEW PT 8RH  H2O(GM.M-3) O3(GM.M-3) NO. DEN.')
0448       106    FORMAT (//10X,29H HORIZONTAL PATH, ALTITUDE =,F7.3,11H KM,RANGE =,
                 1F7.3,3H KM)
0449       107    FORMAT (//10X,50H SLANT PATH BETWEEN ALTITUDES H1 AND H2 WHERE H1
                 1=,F7.3,8H KM H2 =,F7.3,19H KM,ZENITH ANGLE =,F7.3,8H DEGREES)
0450       108    FORMAT (//10X,39H SLANT PATH TO SPACE FROM ALTITUDE H1 =,F7.3,19H
                 1KM, ZENITH ANGLE =,F7.3,8H DEGREES)
0451       109    FORMAT (/25X'HAZE MODEL  =',F5.1,' KM VISUAL RANGE AT SEA LEVEL')
0452       110    FORMAT('  FOG CONDITIONS MAY EXIST AT SEA LEVEL FOR THIS VISUAL RA
                 1NGE',/,'  IF SO THEN ASSUME THE TRANSMITTANCE DUE TO FOG IS GIVEN
                 2BY THE TRANSMITTANCE AT 0.55 MICRONS')
0453       111    FORMAT (/20X,18H MODEL ATMOSPHERE ,I1,11H = TROPICAL)
0454       112    FORMAT (/20X,18H MODEL ATMOSPHERE ,I1,21H = MIDLATITUDE SUMMER)
0455       113    FORMAT (/20X,18H MODEL ATMOSPHERE ,I1,21H = MIDLATITUDE WINTER)
0456       114    FORMAT (/20X,18H MODEL ATMOSPHERE ,I1,21H = SUB-ARCTIC SUMMER )
0457       115    FORMAT (/20X,18H MODEL ATMOSPHERE ,I1,21H = SUB-ARCTIC WINTER )
0458       116    FORMAT (/20X,18H MODEL ATMOSPHERE ,I1,21H = 1962 US STANDARD  )
0459       117    FORMAT (/20X,'AEROSOL SCATTERING NOT COMPUTED,IHAZE=0')
0460       118    FORMAT (/20X,18H       HAZE MODEL ,I1,3H = ,A5,13H VISUAL RANGE)
0461       119    FORMAT (/10X,21H FREQUENCY RANGE V1= ,F7.1,13H CM-1 TO V2= ,F7.1,1
                 14H CM-1 FOR DV =,F6.1,9H CM-1   (,F6.2,' - ',F5.2,' MICRONS )')
0462       120    FORMAT (1H1,///10X,20H HORIZONTAL PROFILES/)
0463       121    FORMAT (' STARTING PARAMETERS H1 AND ANGLE HAVE BEEN REDEFINED:H1=
                 1 ',F10.3,'ANGLE =',F10.6)
0464       122    FORMAT (' TRAJECTORY MISSES EARTHS ATMOSPHERE. CLOSEST DISTANCE OF
                 1 APPROACH IS',F10.2,1X,/,1X,'END OF CALCULATION')
0465       123    FORMAT (3X,I3,F6.1,12(E10.3))
0466       124    FORMAT(1H1,///10X,'  VERTICAL PROFILES  ',45X,'PSI',6X,'PHI',6X,'B
                 1ETA',4X,'THETA    RANGE')
0467       125    FORMAT(I3,F6.1,8E10.3,4F9.4,F6.1)
0468       126    FORMAT (' HMIN = ',F10.3)
0469       127    FORMAT (' H2 WAS SET LESS THAN HMIN AND HAS BEEN RESET EQUAL TO
                 1  HMIN I.E. H2 = ',F10.3)
0470       128    FORMAT (' PATH INTERSECTS EARTH - PATH CHANGED TO TYPE 2 WITH H2 =
                 1 0.0 KM')
0471       129    FORMAT (' CHOICE OF TWO PATHS FOR THIS CASE -SHORTEST PATH TAKEN.
                 1 FOR LONGER PATH SET LEN=1.')
0472       130    FORMAT (' CHOICE OF TWO PATHS FOR THIS CASE -LONGEST PATH TAKEN.
                 1 FOR SHORT PATH SET LEN = 0.')
0473              END
```

```
FORTRAN IV G LEVEL  21                POINT              DATE = 79218          16/22/49                 PAGE 0001

0001            SUBROUTINE POINT (X,YN,N,NP,TX,IP)
0002            COMMON /MOD/ Z(34),P(7,34),T(7,34),EH(15,34),WH(7,34),M,NL,RE,CW,C
               1O,PI,WA(7,34)
0003            COMMON /EMI/ IEMISS,KMAX
0004            DIMENSION TX(15)
          C*********************************************************************
          C     SUBROUTINE POINT COMPUTES THE MEAN REFRATIVE INDEX ABOVE AND BELOW
          C     A GIVEN ALTITUDE AND INTERPOLATES EXPONENTIALLY TO DETERMINE THE
          C     EQUIVALENT ABSORBER AMOUNTS AT THAT ALTITUDE.
          C
          C*********************************************************************
          C
          C      X IS THE HEIGHT IN QUESTION
          C      TX(9) AND YN ARE THE MEAN REFRACTIVE INDICES ABOVE AND BELOW X
          C      N IS THE LEVEL INTEGER CORRESPONDING TO X OR THE LEVEL BELOW X
          C      NP =1 IF X COINCIDES WITH MODEL ATMOSPHERE LEVEL ,IF NOT NP = 0
          C      TX(1-8) ARE ABSORBER  AMOUNTS PER KM AT HEIGHT X
          C*********************************************************************
0005            N=NL
0006            NP=0
0007            IF (X.LT.0.0) X=0.
0008            IF (X.GT.Z(NL)) GO TO 4
0009            DO 1 I=1,NL
0010            N=I
0011            IF (X-Z(I)) 2,4,1
0012        1   CONTINUE
0013        2   J2=N
0014            N=N-1
0015            FAC=(X-Z(N))/(Z(J2)-Z(N))
0016            PX1=P(M,N)*(P(M,J2)/P(M,N))**FAC
0017            TX1=T(M,N)*(T(M,J2)/T(M,N))**FAC
0018            WX1=WH(M,N)*(WH(M,J2)/WH(M,N))**FAC
0019            TX(3)=CO*PX1/TX1-4.56E-6*WX1*TX1*CW
0020            TX(2)=CO*P(M,J2)/T(M,J2)-4.56E-6*WH(M,J2)*T(M,J2)*CW
0021            TX(1)=CO*P(M,N)/T(M,N)-4.56E-6*WH(M,N)*T(M,N)*CW
0022            TX(9)=0.5E-6*(TX(2)+TX(3))
0023            YN=0.5E-6*(TX(1)+TX(3))
0024            IF (IP.EQ.0) GO TO 9
0025            DO 3 K=1,KMAX
0026            IF(K.EQ.9) GO TO 3
0027            TX(K)=0.0
0028            IF (EH(K,N).EQ.0.0) GO TO 3
0029            IF (EH(K,N).GT.1000.0) GO TO 3
0030            TX(K)=EH(K,N)*(EH(K,J2)/EH(K,N))**FAC
0031        3   CONTINUE
0032            GO TO 9
0033        4   NP=1
0034            IF (IP.EQ.0) GO TO 6
0035            DO 5 K=1,KMAX
0036        5   TX(K)=EH(K,N)
0037        6   TX(9)=EH(9,N)-1.
0038            YN=0.0
          C*****  CARDS 8 24 AND 50 THROUGH 59 ARE NO LONGER REQUIRED
0039            IF (N.GT.1) YN=EH(9,N-1)-1.0
0040        9   CONTINUE
0041            IF (IP.EQ.1) PRINT 100, X,N,NP,TX(9),YN,IP,(TX(K),K=1,8)
0042            TX(9)=TX(9)+1.
0043            YN=YN+1.
```

```
0044            RETURN
0045      100   FORMAT (/,' FROM POINT: HEIGHT=',F10.4,' KM,N=',I3,',NP=',I2,',REF
               1. INDEX ABOVE & BELOW X=',2E11.4,',IP=',I3,/,12X,'EQUIV. ABSORBER
               2AMOUNTS PER KM AT X=',8E10.3)
0046            END
```

```
          SUBROUTINE PATH(WLAY,WPATH,TBBY)
          COMMON /MO1/ Z(34),P(7,34),T(7,34),FH(15,34),WH(7,34),V,NL,FE,CW,C
         1O,PI,WA(7,34)
          DIMENSION WLAY(34,15),TBBY(68),WPATH(68,15)
          COMMON /EM1/ IEMISS,KMAX,ANGLE,LEN,HMIN,IJ,J1,J2,JMIN,JEXTRA,ITYPE
          COMMON /EM2/ IL,IKMAX  ,LENTOP,NLL,W(15),E(15)
          COMMON /EM3/ H1,H2,NP1,MODEL
          IF (ITYPE.EQ.1) GO TO 10
          IF (ITYPE.EQ.2.AND.H1.EQ.H2) J2=J1
          IF (H2.GT.H1.AND.ANGLE.GT.90..AND.NP1.EQ.1) J1=J1-1
          IF (JEXTRA.EQ.1) J2=J2+1
          IF((ITYPE.EQ.2).AND.(H1.GT.H2).AND.(LENTOP.EQ.1)) J2=J2-1
          IF (ITYPE.EQ.3) J2=NL
          PRINT 100, J1,J2
          PRINT 101
          DO 1 IK=1,68
          TBBY(IK)=0.
          DO 1 K=1,KMAX
          WPATH(IK,K)=0.
    1     CONTINUE
          LEN=0
          NLL=NL-1
          IL=J1+1
          IJ=IL+NLL
          DO 2 K=1,KMAX
          E(K)=0.
    2     CONTINUE
          IF (ANGLE.GT.90.0) GO TO 3
          LEN=1.
          IL=J1-1
          HMIN=1.0E-6
          IJ=NLL
    3     CONTINUE
          DO 9 IK=1,68
          IF (LEN.EQ.0) IL=IL-1
          IF (LEN.EQ.1) IL=IL+1
          IJ=IJ-1
          IF (IL.EQ.0) GO TO 8
          DO 4 K=1,KMAX
          W(K)=E(K)+WLAY(IL,K)
          WPATH(IK,K)=W(K)
    4     CONTINUE
          IF (IL.LE.0.OR.IL.GE.NL) GO TO 5
          TRAP=(T(M,IL)+T(M,IL+1))*0.5
    C     JEXTRA IS 1 ONLY WHEN PROGRAM NEVER LEAVES ONE LAYER
          IF (JEXTRA.EQ.1) TRAP=(T(M,J1)+T(M,J1+1))*0.5
    5     CONTINUE
          TBBY(IK)=TRAP
          DO 6 K=1,KMAX
          E(K)=W(K)
    6     CONTINUE
          IF (ANGLE.LE.90.0.AND.IL.EQ.NLL) GO TO 9
          IF (ITYPE.EQ.3.AND.ANGLE.LE.90.0) GO TO 7
          IF (ITYPE.EQ.3.AND.LEN.EQ.1.AND.IL.EQ.J2) GO TO 9
          IF((ITYPE.EQ.2.AND.LENTOP.EQ.0.AND.IL.EQ.J2) GO TO 9
          IF (IL.EQ.JMIN.AND.HMIN.GT.0) LEN=1
          IF (IL.EQ.1.AND.HMIN.LE.0.0) GO TO 9
          IF (LEN.EQ.0) GO TO 7
```

```
0057          IF (IL.EQ.JMIN.AND.IJ.EQ.IL+NLL) IL=IL-1
0058          IF (ITYPE.EQ.2.AND.IL.EQ.J2) GO TO 9
0059     7    CONTINUE
0060          PRINT102,IK,(WPATH(IK,K),K=1,8),WPATH(IK,10),WPATH(IK,11),WPATH(IK
             1,12),TRAV(IK)
0061     8    CONTINUE
0062          IKMAX=68
0063          LEN=LENTOR
0064          RETURN
0065     9    CONTINUE
0066          PRINT102,IK,(WPATH(IK,K),K=1,8),WPATH(IK,10),WPATH(IK,11),WPATH(IK
             1,12),TRAV(IK)
0067          IKMAX=IK
0068          LEN=LENTOR
0069          RETURN
0070    10    DO 11 K=1,KMAX
0071          WPATH(1,K)=W(K)
0072    11    CONTINUE
0073          IF (MODEL.EQ.0) J1=1
0074          J2=J1
0075          TRAV(1)=T(M,J1)
0076          IKMAX=1
0077          PRINT 100, J1,J2
0078          PRINT 101
0079          IK=1
0080          PRINT102,IK,(WPATH(IK,K),K=1,8),WPATH(IK,10),WPATH(IK,11),WPATH(IK
             1,12),TRAV(IK)
0081          HMIN=1.0E-6
0082          RETURN
0083   100    FORMAT(9I3)
0084   101    FORMAT (//,20X,' CUMULATIVE ABSORBER AMOUNTS FOR THE ATMOSPHERIC P
             1ATH',//
             210X,'H2O',6X,'CO2+',8X,'O3',9X,'N2',8X,'H2O C',6X,'MOL S',7X,'AERO
             3',6X,'O3 UV',7X,'H2O C',7X,'HNO3',6X,'SO2',6X,'TAVE')
0085   102    FORMAT(I5,10E11.3,F9.3,F8.3)
0086          END
```

```
FORTRAN IV G LEVEL  21              ANGL            DATE = 79218        16/22/49             PAGE 0001

0001                 SUBROUTINE ANGL (H1,H2,ANGLE,B1,LEN,ML)
0002                 COMMON /MOI/ Z(34),P(7,34),T(7,34),EH(15,34),WH(7,34),M,ML,RE,CW,C
                    1O,PI,WA(7,34)
0003                 COMMON /EMI/ IEMISS,KMAX
0004                 DIMENSION TX(15)
           C*****************************************************************
           C
           C         THIS SUBROUTINE CALCULATES THE INITIAL ZENITH ANGLE (ANGLE)
           C         TAKING INTO ACCOUNT REFRACTION EFFECTS GIVEN H1,H2, AND BETA
           C         (WHERE BETA IS THE EARTH CENTRE ANGLE SURTENDED BY H1 AND H2 ),
           C         ASSUMING THE REFRACTIVE INDEX TO BE CONSTANT IN A GIVEN LAYER.
           C         FOR GREATER ACCURACY INCREASE THE NUMBER OF LEVELS IN THE MODEL
           C         ATMOSPHERE.
           C
           C         THIS SUBROUTINE CAN BE REMOVED FROM THE PROGRAM IF NOT REQUIRED.
           C*****************************************************************
0005                 IP=99
0006                 CA=PI/180.
0007                 X1=RE+H1
0008                 X2=RE+H2
0009                 LEN=0.
0010                 IT=0
0011                 B1=B1*CA
0012                 IF(B1.EQ.0.0) B1=ARCOS(X2/X1)
0013                 TANG=X2*SIN(B1)/(X2*COS(B1)-X1)
0014                 THET=ATAN(TANG)
0015                 IF (THET.LT.0.0) THET=THET+PI
0016                 SPHI=SIN(THET)
0017                 ANG=THET/CA
           C         PRINT 404, B1,ANG,TANG
0018                 TN=THET
0019                 TN=TN-0.5*CA
0020       1         ANGLE=THET
0021                 FBT=0.
0022                 BETA=0.
0023                 BET1=0
0024                 BET2=0
0025                 FBT1=0
0026                 FBT2=0
0027                 FBT3=0.0
0028                 IF(B1.LE.0.0) GO TO 2
           C         PRINT 400, IT
0029                 Y=2.*THET
0030                 IF (Y-PI.LT.1.0E-8) GO TO 9
0031                 IF (IP.EQ.100) GO TO 6
0032                 XMIN=X2*COS(B1)-RE
0033                 IF (XMIN-H1) 8,4,4
0034           2     HMIN=H2
0035                 H2=H1
0036                 H1=HMIN
0037       3         ANGLE=0.5*PI
0038                 THET=ANGLE
0039                 SPHI=1.0
0040                 ANG=ANGLE/CA
           C         PRINT 404, B1,ANG,SPHI
0041       4         IP=100
0042                 CALL POINT (H1,YN,N,NP,TX,IP)
0043                 J1=N
```

```
0044            TX1=TX(9)
0045      5     CALL POINT (H2,YN,N,NP,TX,IP)
0046            IF (NP.EQ.1) N=N-1
0047            J2=N
0048            IF (J1.EQ.J2) TX1=TX1+YN-EH(9,J1)
0049      6     DO 7 J=J1,J2
0050            X1=RE+Z(J)
0051            X2=RE+Z(J+1)
0052            IF (J.EQ.J1) X1=RE+H1
0053            IF (J.EQ.J2) X2=RE+H2
0054            SALP=X1*SPHI/X2
0055            ALP=ARSIN(SALP)
0056            RN=EH(9,J+1)/EH(9,J)
0057            IF ((J+1).EQ.J2) RN=YN/EH(9,J)
0058            IF (J.EQ.J1) RN=EH(9,J+1)/TX1
0059            IF ((J+1).EQ.J2.AND.J.EQ.J1) RN=YN/TX1
0060            BET=THET-ALP
0061            FB=-TAN(ALP)
0062            IF (J.NE.J1) FB=FB+TAN(THET)
0063            FBT=FBT+FB
0064            BETA=BETA+BET
0065            TH1=THET/CA
0066            BE=BET/CA
0067            C=ALP/CA
          C     PRINT 402, J,Z(J),THET,ALP,BET,BETA,FBT,FB,TH1,BE,C
0068            IF (X2.EQ.RE+H2) C=PI-ALP
0069            IF (SALP.GE.RN) RN=1.
0070            SPHI=SALP/RN
0071            THET=ARSIN(SPHI)
0072      7     CONTINUE
0073            IF(B1.LE.0.0) GO TO 29
0074            GO TO 26
0075      8     CONTINUE
0076            TANG=-TANG
0077            ANGLE=PI-ANGLE
0078            TN=ANGLE
0079            ANG=ANGLE/CA
          C     PRINT 404, B1,ANG,TANG
0080            IF (H1.LE.0.0) GO TO 3
0081      9     CONTINUE
0082            IP=101
0083            CALL POINT (H1,YN,N,NP1,TX,IP)
0084            TX1=TX(9)
0085            YN1=YN
0086            IF (NP1.EQ.1) N=N-1
0087            J2=ML
0088            IF (M.EQ.7) J2=ML
0089            J1=N
0090            J=J1+1
0091            IF (H2.GE.H1) GO TO 13
0092            CALL POINT (H2,YN,N,NP,TX,IP)
0093            TX2=TX(9)
0094            YN2=YN
0095            J2=N
0096            IF (J1.EQ.J2) TX2=YN1+TX(9)-EH(9,J1)
0097     10     J=J-1
0098            X1=RE+Z(J+1)
0099            X2=RE+Z(J)
```

```
      IF (J.EQ.J1) X1=RE+H1
      IF (J.EQ.J2) X2=RE+H2
      SALP=X1*SPHI/X2
      HMIN=X1*SPHI-RE
C     PRINT 402, J,X1,Z(J),SPHI,SALP,HMIN,RE
      IF (SALP.LE.1.0) GO TO 11
      SALP=SPHI
      IF (HMIN.GT.H2) GO TO 18
11    ALP=ARSIN(SALP)
      THET=ARSIN(SPHI)
      BET=ALP-THET
      BET1=BET1+BET
      FB=TAN(ALP)
      IF (J.NE.J1) FB=FB-TAN(THET)
      FBT1=FBT1+FB
      TH1=THET/CA
      BE=BET/CA
      AL=ALP/CA
C     PRINT 402, J,X2,THET,ALP,BET1,BET,BMIN,HMIN,FBT1,TH1,BE,AL
      IF (X2.EQ.RE+H2) C=PI-ALP
      REF=EH(9,J)
      IF (J.EQ.J1) REF=YN1
      IF (J.EQ.J2) REF=TX2
      IF (J.EQ.1) GO TO 12
      RN=EH(9,J)/EH(9,J-1)
      IF (J.EQ.J1) RN=YN1/EH(9,J-1)
      IF(J.EQ.J2+1) RN=REF/TX2
      IF(J.EQ.J2) RN=REF/YN2
      IF (SALP.GE.RN) RN=1.
      SPHI=SALP*RN
      IF (Z(J).LE.H2) GO TO 12
      GO TO 10
12    X1=X2
      IF (ABS(Z(J)-H2).LT.1.0E-10.AND.J.NE.1) GO TO 13
      GO TO 14
13    J=J-1
      X1=RE+Z(J+1)
      IF (J.EQ.J1) X1=RE+H1
      IF (J.EQ.J2.AND.J.NE.J1) X1=RE+H2
14    X2=RE+Z(J)
      HMIN=X1*SPHI-RE
      IF (HMIN.LE.0.0) GO TO 25
      IF (Z(J).LT.HMIN) GO TO 18
      REF=EH(9,J)
      IF (J.EQ.J2) REF=YN
      SALP=X1*SPHI/X2
      ALP=ARSIN(SALP)
      THET=ARSIN(SPHI)
      BET=ALP-THET
      FB=TAN(ALP)-TAN(THET)
      FBT2=FBT2+FB
      BET2=BET2+BET
      BMIN=BET1+BET2
      AL=ALP/CA
      TH1=THET/CA
C     PRINT 402, J,X2,THET,ALP,BET2,BET,BMIN,HMIN,FBT2,TH1,BE,AL
      RN=REF/EH(9,J-1)
      IF (SALP.GE.RN) RN=1.0
```

```
      SPHI=SALP*RN
      GO TO 13
   17 TX3=YN1+TX(9)-EH(9,J1)
      YN1=TX3
      IF (ABS(H2-Z(J+1)).LE.1.0E-5) YN1=TX(9)
      IF (ABS(H1-Z(J+1)).LE.1.0E-5) YN1=TX(9)
      RN=1.0
      GO TO 19
   18 CALL POINT (HMIN,YN,N,NP,TX,IF)
      IP=102
      TX3=TX(9)
      IF (J.EQ.J1.AND.H2.GE.H1) GO TO 17
      IF (J.EQ.J1.OR.J.EQ.J2) TX3=YN2+TX(9)-EH(9,J)
      IF (HMIN.GT.H2) TX3=TX(9)
      IF (J.EQ.J1.AND.HMIN.GT.H2) GO TO 17
      RN=REF/TX3
      IF (SALP.GE.RN) RN=1.
      SPHI=SALP*RN
      X=X1*SPHI-RE
      DIF=ABS(HMIN-X)
      HMIN=X
      IF (DIF-1.0E-5) 19,19,18
   19 X2=RE+HMIN
C     PRINT 403, HMIN,DIF,RN
      THET=ARSIN(SPHI)
      IF(RN.EQ.1.0) FBT3=-TAN(THET)
      IF (RN.EQ.1) GO TO 20
      DNX=(TX3-1.0)*ALOG((TX3-1.0)/(REF-1.0))/(X2-X1)
      FBT3=-TAN(THET)*(1.0-1.0/(1.0+TX3/(X2*DNX)))
   20 BET=0.5*PI-THET
      BET2=BET2+BET
      BMIN=BET1+BET2
      IF (H2.GE.H1) GO TO 23
      BET=BET1+2.*BET2
      DB1=B1-BET1
      DB2=BET-B1
   21 DB3=ABS(BMIN-B1)
      IF(DB3.GT.DB1.AND.DB2.GT.DB1) GO TO 25
      IF(DB2.GT.DB3) GO TO 22
      IF(DB2.GT.DB1)  GO TO 25
      BETA=BET
      FBT=FBT1+2.0*(FBT2+FBT3)
      LEN=1.
      GO TO 26
   22 BETA=BET1+BET2
      FBT=FBT1+FBT2+FBT3
C     PRINT 401, J,BETA,FBT,FBT1,FBT2,FBT3,TX1,YN1
      GO TO 26
   23 BETA=2.0*(BET1+BET2)
      LEN=1.
      FBT=2.0*(FBT1+FBT2+FBT3)
      PRINT 401, J,BETA,FBT,FBT1,FBT2,FBT3,TX1,YN1
      IF (H2.EQ.H1) GO TO 26
      IP=103
      IF (NP1.EQ.1) J1=J1+1
      SPHI=SIN(ANGLE)
      IF (Z(J1+1).LE.H2) GO TO 24
      RN=TX1/YN1
```

```
            IF (SPHI.GE.RN) RN=1.
            SPHI=SPHI/RN
            THET=ARSIN(SPHI)
            GO TO 5
      24    CALL POINT (H2,YN,N,NP,TX,IP)
            TX1=TX1+YN-FH(9,J1)
            RN=TX1/YN1
            J2=J1
            IF (SPHI.GE.RN) RN=1.
            SPHI=SPHI/RN
            THET=ARSIN(SPHI)
            GO TO 5
      25    BETA=BET1
            LEN=0.
            FBT=FBT1
      26    THET=ANGLE+(B1-BETA)/(1.+FBT/TANG)
            DBETA=BETA/CA
            B=BET1/CA
            TH1=THET/CA
            PRINT 404, BETA,DBETA,FBT,TH1,TANG
            IF (THET.GT.TN.OR.THET.LT.TM) THET=(TN+TM)/2.
            TH1=THET/CA
            PRINT 404, BET1,B,FBT,TH1
            TN1=TN/CA
            TM1=TM/CA
            PRINT 405, TN,TM,TN1,TM1
            SPHI=SIN(THET)
            TANG=TAN(THET)
            IT=IT+1
            DBE=ABS(B1-BETA)
            DTH=ABS(ANGLE-THET)
            IF (IT.EQ.10)  THET=0.5*(ANGLE+THET)
            IF (IT.EQ.10) GO TO 28
            IF (DBE.GT.1.0E-7.AND.DTH.GT.1.0E-7) GO TO 1
      28    ANGLE=THET/CA
            PRINT 406, ANGLE,IT
            RETURN
         29 H1=H2
            ANGLE=C/CA
            PRINT 406, ANGLE,IT
            RETURN
      C
      400   FORMAT (//' ITTERATION NUMBER ',I3,//)
      401   FORMAT (I6,F16.7,8F13.8)
      402   FORMAT (I4,F10.4,6F13.4,4F10.4/)
      403   FORMAT('  HMIN=',F14.6,'  DIF=',F14.6,'  P2=',F16.8)
      404   FORMAT (' TOTAL BETA = ',E14.6,F15.6,',FBT = ',F14.6,' THET =',F10
           1.6,'TANG=',F10.6)
      405   FORMAT (5F12.6)
      406   FORMAT (8X,/1H','ZENITH ANGLE =',F7.3,' DEGREES :  RECOMPUTED
           1 FROM SUBROUTINE ANGL (ITTERATION',I3,')')
            END
```

```
FORTRAN IV G LEVEL  21                FREQSL            DATE = 79218          16/22/49              PAGE 0001

0001                SUBROUTINE FREQSL(I,IV,W,IHAZE,TX,ALAM)
0002                COMMON /M05/ C1(2580),C2(1575),C3(540),C4(133),C5(15),C8(102),C11(
                   1144),C12(115),C13(43),C14(109),C15(85)
0003                COMMON /M06/ VX(45),C7(45),C7A(45)
0004                COMMON /M07/ TR(67),FW(67),FO(67)
0005                COMMON /M08/ SUM4,SUM5,SUM6,SUM7,SUM8,SUM11
0006                COMMON /M010/ FS(9),S1(9),S2(9)
0007                COMMON /M011/ FN3(9),FN1(9),FN2(9)
0008                COMMON /M012/ FNH3(9),FH1(9),FH2(9)
0009                COMMON /M013/ FNO2(9),O1(9),O2(9)
0010                DIMENSION W(15),TX(15)
              C*******************************************************************
              C
              C     THIS SUBROUTINE SELECTS THE ATTENUATOR EFFECTIVE AT THE FREQUENCY
              C     INPUT FROM 13 MODEL SUBROUTINE.
              C
              C*******************************************************************
0011                IF (I.GE.01.AND.I.LE.18) GO TO 4
0012                IF (I.GE.19.AND.I.LE.30) GO TO 3
0013                IF (I.GE.31.AND.I.LE.45) GO TO 2
0014                IF (I.GE.46.AND.I.LE.54) GO TO 1
0015                IF (I.GE.55.AND.I.LE.60) GO TO 6
0016                IF (I.GE.61.AND.I.LE.64) GO TO 8
0017                IF ((I.GE.65.AND.I.LE.68).OR.(I.GE.499.AND.I.LE.510)) GO TO 9
0018                IF (I.GE.69.AND.I.LE.100) GO TO 10
0019                IF (I.GE.101.AND.I.LE.107) GO TO 11
0020                IF (I.GE.108.AND.I.LE.141) GO TO 12
0021                IF (I.GE.142.AND.I.LE.177) GO TO 15
0022                IF ((I.GE.178.AND.I.LE.181).OR.(I.GE.193.AND.I.LE.212)) GO TO 16
0023                IF ((I.GE.182.AND.I.LE.192).OR.(I.GE.213.AND.I.LE.238).OR.(I.GE.26
                   16.AND.I.LE.277)) GO TO 13
0024                IF (I.GE.239.AND.I.LE.265) GO TO 17
0025                IF ((I.GE.278.AND.I.LE.282).OR.(I.GE.326.AND.I.LE.346).OR.(I.GE.48
                   10.AND.I.LE.498).OR.(I.GE.511.AND.I.LE.531)) GO TO 14
0026                IF (I.GE.283.AND.I.LE.325) GO TO 18
0027                IF ((I.GE.347.AND.I.LE.420).OR.(I.GE.439.AND.I.LE.479)) GO TO 19
0028                IF (I.GE.421.AND.I.LE.438) GO TO 20
0029                IF (I.GE.531.AND.I.LE.585) GO TO 21
0030                IF (I.GE.586.AND.I.LE.1545) GO TO 22
0031                IF ((I.GE.1546.AND.I.LE.1770).OR.(I.GE.1906.AND.I.LE.2490)) GO TO
                   123
0032                IF ((I.GE.1771.AND.I.LE.1905).OR.(I.GE.2491.AND.I.LE.2520).OR.(I.G
                   1E.4612.AND.I.LE.5430)) GO TO 24
0033                IF (I.GE.2521.AND.I.LE.2530) GO TO 25
0034                IF (I.GE.2531.AND.I.LE.2580) GO TO 26
0035                IF((I.GE.2581.AND.I.LE.2610).OR.(I.GE.2836.AND.I.LE.4611).OR.(I.GE
                   1.5431)) GO TO 27
0036                IF (I.GE.2611.AND.I.LE.2835) GO TO 28
0037          1     CALL EVETS(I,W,C3,TX)
0038          2     CALL DIVAD(I,W,C2,TX)
0039          3     CALL SUSEJ(I,W,C12,TX)
0040          4     CALL LUAP(I,W,C1,TX)
0041          5     CALL FKUL(I,IV,W,IHAZE,TX,SUM7,ALAM)
0042                GO TO 40
0043          6     CALL EVETS(I,W,C3,TX)
0044          7     CALL DIVAD(I,W,C2,TX)
0045                GO TO 4
0046          8     CALL ROJ(I,W,C15,TX)
```

```
FORTRAN IV G LEVEL  21            FREQSL            DATE = 79218          16/22/49            PAGE 0002

                 GO TO 6
          9      CALL NHOJ(I,IV,W,C5,TX,SUM5)
                 GO TO 8
         10      CALL SUTIT(I,W,C14,TX)
                 GO TO 9
         11      CALL RETEP(I,W,C11,SUM11,TX)
                 GO TO 10
         12      CALL SUTIT(I,W,C14,TX)
         13      CALL RETEP(I,W,C11,SUM11,TX)
         14      CALL NHOJ(I,IV,W,C5,TX,SUM5)
                 GO TO 6
         15      CALL SUSEJ(I,W,C12,TX)
                 GO TO 12
         16      CALL SUSEJ(I,W,C12,TX)
                 GO TO 13
         17      CALL BOJ(I,W,C15,TX)
                 GO TO 13
         18      CALL ARZE(I,W,C13,TX)
                 GO TO 14
         19      CALL KRAM(I,W,C4,TX,SUM4)
                 GO TO 14
         20      CALL SUSEJ(I,W,C12,TX)
                 GO TO 19
         21      CALL SEMAJ(I,IV,W,TX,SUM6)
                 GO TO 6
         22      CALL SEMAJ(I,IV,W,TX,SUM6)
                 GO TO 7
         23      CALL SEMAJ(I,IV,W,TX,SUM6)
                 GO TO 4
         24      CALL SEMAJ(I,IV,W,TX,SUM6)
                 GO TO 5
         25      CALL DIVAD(I,W,C2,TX)
                 GO TO 24
         26      CALL SESOM(I,W,C8,TX,SUM8)
                 GO TO 25
         27      CALL SESOM(I,W,C8,TX,SUM8)
                 GO TO 24
         28      CALL LUAP(I,W,C1,TX)
                 GO TO 27
         40      RETURN
                 END
```

```
FORTRAN IV G LEVEL  21                LUAP              DATE = 79218         16/22/49              PAGE 0001

            SUBROUTINE LUAP(I,W,C1,TX)
            DIMENSION C1(2580),TX(1),WS(1),W(1)
      C*********************************************************************
      C
      C     TRANSMITTANCE FOR WATER VAPOR
      C
      C     THIS SUBROUTINE USES A CONTINUOUS FUNCTION FOR THE ORIGINAL
      C     TRANSMITTANCE TABLE.
      C
      C*********************************************************************
            IF (W(1).LT.1.0E-20) GO TO 5
            IF (I.LE.1170) I1=I
            IF (I.GT.1905.AND.I.LE.2490) I1=I-135
            IF (I.GT.2610) I1=I-255
            WS(1)=ALOG10(W(1))+C1(I1)
            TX(1)=EXP(-10**(-1.14619+0.55013*WS(1)))
      5     RETURN
            END
```

```
FORTRAN IV G LEVEL  21                DIVAD               DATE = 79218        16/22/49            PAGE 0001

0001              SUBROUTINE DIVAD(I,W,C2,TX)
0002              DIMENSION C2(1575),TX(2),WS(2),W(2)
            C*****************************************************************
            C
            C     TRANSMITTANCE FOR UNIFORMLY MIXED GASES
            C
            C     THIS SUBROUTINE USES A CONTINUOUS FUNCTION FOR THE ORIGINAL
            C     TRANSMITTANCE TABLE.
            C
            C*****************************************************************
0003              IF (W(2).LT.1.0E-20) GO TO 5
0004              I1=I-30
0005              IF (I.GT.2520) I1=I-1905
0006              WS(2)=ALOG10(W(2))+C2(I1)
0007              TX(2)=EXP(-10**(-1.14619+0.55C13*WS(2)))
0008        5     RETURN
0009              END
```

```
FORTRAN IV G LEVEL  21              EVETS              DATE = 79218         16/22/49              PAGE 0001

0001             SUBROUTINE EVETS(I,W,C3,TX)
0002             DIMENSION C3(540),TX(3),WS(3),W(3)
           C**********************************************************************
           C
           C      TRANSMITTANCE FOR OZONE
           C
           C      THIS SUBROUTINE USES A CONTINUOUS FUNCTION FOR THE ORIGINAL
           C      TRANSMITTANCE TABLE.
           C
           C**********************************************************************
0003             IF (W(3).LT.1.0E-20) GO TO 5
0004             I1=I-45
0005             WS(3)=ALOG10(W(3))+C3(I1)
0006             TX(3)=1/(1+EXP(-3.08319+2.11127*WS(3)))
0007       5     RETURN
0008             END
```

```
FORTRAN IV G LEVEL  21                NHOJ              DATE = 79218        16/22/49            PAGE 0001

              SUBROUTINE NHOJ(I,IV,W,C5,TX,SUMS)
        C******TRANSMITTANCE FOR WATER VAPOR CONTINUM 4 MICRON AND 10 MICRON REGION
              DIMENSION C5(15),TX(5),W(10)
              IF(I.LT.65.OR.I.GT.531) GO TO 2
              IF(I.GT.201) GO TO 1
              TX(5)=(4.18+5578.0*EXP(-7.87E-3*IV))*W(5)
              GO TO 3
         1    IF(I.LT.401) GO TO 2
              XI=(I-401.0)/10.0+1.0
              NH=XI+1.001
              XH=XI-FLOAT(NH)
              TX(5)=C5(NH)
              TX(5)=TX(5)+XH*(C5(NH)-C5(NH-1))
              TX(5)=TX(5)*W(10)
              GO TO 3
         2    TX(5)=0.0
         3    SUMS=TX(5)
              IF (TX(5).EQ.0.0) GO TO 5
              IF (TX(5).LE.0.1) GO TO 4
              IF (TX(5).GT.20.) GO TO 6
              TX(5)=EXP(-TX(5))
              GO TO 7
         4    TX(5)=1.0-TX(5)+0.5*TX(5)*TX(5)
              GO TO 7
         5    TX(5)=1.0
              GO TO 7
         6    TX(5)=0.0
         7    RETURN
              END
```

```
FORTRAN IV G LEVEL  21                KRAM              DATE = 79218         16/22/49              PAGE 0001

                 SUBROUTINE KRAM(I,W,C4,TX,SUM4)
           C*********TRANSMITTANCE FOR NITROGEN CONTINUUM****************************
                 DIMENSION C4(133),TX(4),W(4)
                 IF (I.LT.307) GO TO 4
                 I1=I-346
                 TX(4)=C4(I1)*W(4)
                 SUM4=TX(4)
                 IF (TX(4).EQ.0.0) GO TO 4
                 IF (TX(4).LE.0.1) GO TO 3
                 IF (TX(4).GT.20.) GO TO 5
                 TX(4)=EXP(-TX(4))
                 GO TO 6
               3 TX(4)=1.0-TX(4)+0.5*TX(4)*TX(4)
                 GO TO 6
               4 TX(4)=1.0
                 GO TO 6
               5 TX(4)=0.0
               6 RETURN
                 END
```

```
FORTRAN IV G LEVEL  21                SEMAJ              DATE = 79218         16/22/49              PAGE 0001

              SUBROUTINE SEMAJ(I,IV,W,TX,SUM6)
        C*********TRANSMITTANCE FOR MOLECULAR SCATTERING ***********************
              DIMENSION TX(6),W(6)
              V=IV
              C6=9.807E-20*(V**4.0117)
              TX(6)=C6*W(6)
              SUM6=TX(6)
              IF (TX(6).EQ.0.0) GO TO 4
              IF (TX(6).LE.0.1) GO TO 3
              IF (TX(6).GT.20.) GO TO 5
              TX(6)=EXP(-TX(6))
              GO TO 6
        3     TX(6)=1.0-TX(6)+0.5*TX(6)*TX(6)
              GO TO 6
        4     TX(6)=1.0
              GO TO 6
        5     TX(6)=0.0
        6     RETURN
              END
```

```
FORTRAN IV G LEVEL  21                 EKUL                DATE = 79218        16/22/49              PAGE 0001

              SUBROUTINE EKUL(I,IV,W,IHAZE,TX,SUM7,ALAM)
        C******** TRANSMITTANCE FOR AEROSOL ************************************
              COMMON /MO6/ VX(45),C7(45),C7A(45)
              DIMENSION TX(10),W(7)
              V=IV
              ALAM=1.0E+4/V
              XX=0.0
              YY=0.0
              IF(IHAZE.EQ.0) GO TO 3
              DO 1 N=1,44
              XD=ALAM-VX(N)
              IF(XD) 2,1,1
        1     CONTINUE
        2     XX=(C7(N)-C7(N-1))*XD/(VX(N)-VX(N-1))+C7(N)
        3     TX(7)=XX*W(7)
              SUM7=TX(7)
              IF (TX(7).EQ.0.0) GO TO 5
              IF (TX(7).LE.0.1) GO TO 4
              IF (TX(7).GT.20.) GO TO 6
              TX(7)=EXP(-TX(7))
              GO TO 7
        4     TX(7)=1.0-TX(7)+0.5*TX(7)*TX(7)
              GO TO 7
        5     TX(7)=1.0
              GO TO 7
        6     TX(7)=0.0
        7     CONTINUE
              IF(IHAZE.EQ.0) GO TO 12
              YY=(C7A(N)-C7A(N-1))*XD/(VX(N)-VX(N-1))+C7A(N)
        12    TX(10)=YY*W(7)
              IF (TX(10).EQ.0.0) GO TO 9
              IF (TX(10).LE.0.1) GO TO 8
              IF (TX(10).GT.20.) GO TO 10
              TX(10)=EXP(-TX(10))
              GO TO 11
        8     TX(10)=1.0-TX(10)+0.5*TX(10)*TX(10)
              GO TO 11
        9     TX(10)=1.0
              GO TO 11
        10    TX(10)=0.0
        11    RETURN
              END
```

```
FORTRAN IV G LEVEL  21                SESOM             DATE = 79218          16/22/49             PAGE 0001

                SUBROUTINE SESOM(I,W,CB,TX,SUMB)
          C******** TRANSMITTANCE FOR UV OZONE ***********************************
                DIMENSION CB(102),TX(8),W(8)
                AI=I
                IF(I.LE.4611) GO TO 1
                IF(I.GE.5431) GO TO 2
          1     XX=40.0
                XI=(AI-2531.0)/XX+1.0
                L1=1
                L2=53
                GO TO 3
          2     XX=100.0
                XI=(AI-5431.0)/XX+57.0
                L1=57
                L2=102
          3     DO 4  N=L1,L2
                XD=XI-FLOAT(N)
                IF (XD) 6,5,4
          4     CONTINUE
          5     TX(8)=W(8)*CB(N)
                GO TO 8
          6     TX(8)=CB(N)+XD*(CB(N)-CB(N-1))
                TX(8)=W(8)*TX(8)
          8     SUMB=TX(8)
                IF (TX(8).EQ.0.0) GO TO 10
                IF (TX(8).LE.0.1) GO TO 9
                IF (TX(8).GT.20.0) GO TO 11
                TX(8)=EXP(-TX(8))
                GO TO 12
          9     TX(8)=1.0-TX(8)+0.5*TX(8)*TX(8)
                GO TO 12
          10    TX(8)=1.0
                GO TO 12
          11    TX(8)=0.0
          12    RETURN
                END
```

```
FORTRAN IV G LEVEL  21            RETEP            DATE = 79218        16/22/49            PAGE 0001

          SUBROUTINE RETEP(I,W,C11,TX,SUM11)
C********* TRANSMITTANCE FOR NITRIC ACID**********************************
          DIMENSION C11(44),TX(11),W(11)
          HABS=0.
          IF(I.LT.100.OR.I.GT.276) GO TO 1
          IF(I.GT.116.AND.I.LT.186) GO TO 1
          IF(I.GT.201.AND.I.LT.266) GO TO 1
          IF (I.LE.116) I1=I-100
          IF (I.GE.186.AND.I.LE.201) I1=I-170
          IF (I.GE.266) I1=I-234
          HABS=C11(I1)
    1     CONTINUE
          TX(11)=HABS*W(11)
          SUM11=TX(11)
          IF (TX(11).EQ.0.0) GO TO 6
          IF (TX(11).LE.0.1) GO TO 5
          IF (TX(11).GT.20.) GO TO 7
          TX(11)=EXP(-TX(11))
          GO TO 8
    5     TX(11)=1.0-TX(11)+0.5*TX(11)*TX(11)
          GO TO 8
    6     TX(11)=1.0
          GO TO 8
    7     TX(11)=0.0
    8     RETURN
          END
```

```
FORTRAN IV G LEVEL  21                SUSEJ             DATE = 79218          16/22/49              PAGE 0001

0001                SUBROUTINE SUSEJ(I,W,C12,TX)
0002                COMMON /MOLO/ FS(9),S1(9),S2(9)
0003                DIMENSION C12(115),TX(12),WS(12),W(12)
          C*********************************************************************
          C
          C         THIS SUBROUTINE CALCULATES THE TRANSMITTANCE BY SO2 ( PPM READ IN
          C         THE MAIN PROGRAM).
          C
          C*********************************************************************
0004                IF (W(12).LT.1.0E-20) GO TO 5
0005                IF (I.GE.19.AND.I.LE.54) I1=I-18
0006                IF (I.GE.142.AND.I.LE.181) I1=I-104
0007                IF (I.GE.193.AND.I.LE.213) I1=I-116
0008                IF (I.GE.421) I1=I-323
0009                WS(12)=ALOG10(W(12))+C12(I1)
0010                DO 1 J=1,9
0011                IF(WS(12)-FS(J)) 2,2,1
0012          1     CONTINUE
0013          2     TX(12)=EXP(-10**(S1(J)+S2(J)*WS(12)))
0014          5     RETURN
0015                END
```

```
FORTRAN IV G LEVEL  21              ARZE              DATE = 79218          16/22/49               PAGE 0001

                  SUBROUTINE ARZE(I,W,C13,TX)
                  COMMON /MO11/ FN0(9),FN1(9),FN2(9)
                  DIMENSION C13(43),TX(13),WS(13),W(13)
            C*********************************************************************
            C
            C     THIS SUBROUTINE CALCULATES THE TRANSMITTANCE BY NO  ( PPM READ IN
            C     THE MAIN PROGRAM).
            C
            C*********************************************************************
                  I1=I-282
                  IF (W(13).LT.1.0E-20) GO TO 3
                  WS(13)=ALOG10(W(13))+C13(I1)
                  DO 1 J=1,9
                  IF (WS(13)-FN0(J)) 2,2,1
            1     CONTINUE
            2     TX(13)=EXP(-10**(FN1(J)+FN2(J)*WS(13)))
            3     RETURN
                  END
```

```
FORTRAN IV G LEVEL  21                SUTIT              DATE = 79218        16/22/49              PAGE 0001

 0001                 SUBROUTINE SUTIT(I,W,C14,TX)
 0002                 COMMON /MO12/ FNH3(9),FH1(9),FH2(9)
 0003                 DIMENSION C14(109),TX(14),WS(14),W(14)
            C*********************************************************************
            C
            C         THIS SUBROUTINE CALCULATES THE TRANSMITTANCE BY NH3 ( PPM READ IN
            C         THE MAIN PROGRAM).
            C
            C*********************************************************************
 0004                 I1=I-68
 0005                 IF (W(14).LT.1.0E-20) GO TO 3
 0006                 WS(14)=ALOG10(W(14))+C14(I1)
 0007                 DO 1 J=1,9
 0008                 IF (WS(14)-FNH3(J)) 2,2,1
 0009           1     CONTINUE
 0010           2     TX(14)=EXP(-10**(FH1(J)+FH2(J)*WS(14)))
 0011           3     RETURN
 0012                 END
```

```
FORTRAN IV G LEVEL  21                BOJ              DATE = 79218        16/22/49              PAGE 0001

                  SUBROUTINE BOJ(I.W.C15.TX)
                  COMMON /MO13/ FNO2(9).O1(9).C2(9)
                  DIMENSION C15(45).TX(15).WS(15).W(15)
            C*********************************************************************
            C
            C     THIS SUBROUTINE CALCULATES THE TRANSMITTANCE BY NO2 ( PPM READ IN
            C     THE MAIN PROGRAM).
            C
            C*********************************************************************
                  IF (I.LE.106) I1=I-105
                  IF (I.GE.239.AND.I.LE.265) I1=I-192
                  IF (I.GE.499.AND.I.LE.510) I1=I-425
                  IF (W(15).LT.1.0E-20) GO TO 3
                  WS(15)=ALOG10(W(15))+C15(I1)
                  DO 1 J=1.9
                  IF (WS(15)-FNO2(J)) 2.2.1
            1     CONTINUE
            2     TX(15)=EXP(-10**(O1(J)+C2(J)*WS(15)))
            3     RETURN
                  END
```

```
PROGRAM WILL BE EXECUTED IN THE TRANSMISSION MODE
1  1  1  0  0  0  0  0  0  0  0       0.0        0.0
              0.0        2.500    65.000     5.000      0.0        0.0
 450.000   455.000      5.000

         HORIZONTAL PATH, ALTITUDE =  0.0   KM,RANGE =  5.000 KM

                   MODEL ATMOSPHERE 1 = TROPICAL

                        HAZE MODEL 1 =  23KM VISUAL RANGE

         FREQUENCY RANGE V1=   450.0 CM-1 TO V2=   455.0 CM-1 FOR DV =   5.0 CM-1   ( 21.98 - 22.22 MICRONS )
```

HORIZONTAL PROFILES

1	0.0	0.182E 01	0.979E 00	0.256E-02	0.695E 00	0.492E-01	0.910E 00	0.100E 01	0.261E-02	0.249E-03	0.255E 00	0.0	0.198E-01
2	1.0	0.114E 01	0.741E 00	0.246E-02	0.571E 00	0.259E-01	0.829E 00	0.440E 00	0.261E-02	0.227E-03	0.164E 00	0.0	0.179E-01
3	2.0	0.738E 00	0.622E 00	0.227E-02	0.467E 00	0.149E-01	0.754E 00	0.190E 00	0.252E-02	0.205E-03	0.112E 00	0.0	0.162E-01
4	3.0	0.338E 00	0.515E 00	0.205E-02	0.376E 00	0.436E-02	0.679E 00	0.798E-01	0.238E-02	0.186E-03	0.513E-01	0.0	0.145E-01
5	4.0	0.143E 00	0.431E 00	0.181E-02	0.306E 00	0.120E-02	0.616E 00	0.422E-01	0.219E-02	0.169E-03	0.233E-01	0.0	0.131E-01
6	5.0	0.883E-01	0.359E 00	0.166E-02	0.248E 00	0.662E-03	0.558E 00	0.318E-01	0.210E-02	0.152E-03	0.158E-01	0.0	0.118E-01
7	6.0	0.451E-01	0.296E 00	0.151E-02	0.199E 00	0.264E-03	0.503E 00	0.224E-01	0.201E-02	0.137E-03	0.874E-02	0.0	0.106E-01
8	7.0	0.224E-01	0.245E 00	0.138E-02	0.159E 00	0.105E-03	0.453E 00	0.208E-01	0.191E-02	0.124E-03	0.485E-02	0.0	0.944E-02
9	8.0	0.107E-01	0.201E 00	0.125E-02	0.127E 00	0.405E-04	0.408E 00	0.215E-01	0.182E-02	0.111E-03	0.261E-02	0.408E-54	0.841E-02
10	9.0	0.459E-02	0.163E 00	0.119E-02	0.999E-01	0.136E-04	0.364E 00	0.206E-01	0.182E-02	0.990E-04	0.124E-02	0.364E-05	0.747E-02
11	10.0	0.171E-02	0.133E 00	0.113E-02	0.789E-01	0.405E-05	0.325E 00	0.201E-01	0.182E-02	0.883E-04	0.528E-03	0.107E-04	0.661E-02
12	11.0	0.516E-03	0.107E 00	0.113E-02	0.616E-01	0.130E-05	0.290E 00	0.188E-01	0.191E-02	0.784E-04	0.184E-03	0.232E-04	0.583E-02
13	12.0	0.161E-03	0.858E-01	0.112E-02	0.476E-01	0.278E-06	0.256E 00	0.197E-01	0.201E-02	0.693E-04	0.656E-04	0.308E-04	0.513E-02
14	13.0	0.426E-04	0.630E-01	0.111E-02	0.365E-01	0.676E-07	0.226E 00	0.182E-01	0.210E-02	0.613E-04	0.204E-04	0.317E-04	0.449E-02
15	14.0	0.209E-04	0.543E-01	0.105E-02	0.281E-01	0.320E-07	0.200E 00	0.179E-01	0.210E-02	0.538E-04	0.120E-04	0.320E-04	0.391E-02
16	15.0	0.138E-04	0.422E-01	0.103E-02	0.210E-01	0.206E-07	0.174E 00	0.168E-01	0.219E-02	0.469E-04	0.929E-05	0.314E-04	0.340E-02
17	16.0	0.101E-04	0.327E-01	0.967E-03	0.157E-01	0.148E-07	0.152E 00	0.160E-01	0.219E-02	0.404E-04	0.832E-05	0.289E-04	0.293E-02
18	17.0	0.765E-05	0.247E-01	0.133E-02	0.113E-01	0.110E-07	0.130E 00	0.158E-01	0.322E-02	0.340E-04	0.660E-05	0.259E-04	0.246E-02
19	18.0	0.580E-05	0.177E-01	0.161E-02	0.780E-02	0.823E-08	0.107E 00	0.153E-01	0.420E-02	0.281E-04	0.431E-05	0.225E-04	0.200E-02
20	19.0	0.483E-05	0.128E-01	0.233E-02	0.540E-02	0.680E-08	0.885E-01	0.128E-01	0.653E-02	0.233E-04	0.312E-05	0.203E-04	0.164E-02
21	20.0	0.379E-05	0.937E-02	0.295E-02	0.377E-02	0.528E-08	0.736E-01	0.943E-02	0.887E-02	0.194E-04	0.214E-05	0.221E-04	0.134E-02
22	21.0	0.368E-05	0.686E-02	0.348E-02	0.265E-02	0.512E-08	0.613E-01	0.684E-02	0.112E-01	0.162E-04	0.182E-05	0.227E-04	0.111E-02
23	22.0	0.316E-05	0.505E-02	0.380E-02	0.187E-02	0.437E-08	0.513E-01	0.514E-02	0.131E-01	0.136E-04	0.138E-05	0.215E-04	0.915E-03
24	23.0	0.290E-05	0.380E-02	0.407E-02	0.135E-02	0.400E-08	0.435E-01	0.394E-02	0.149E-01	0.116E-04	0.118E-05	0.226E-04	0.765E-03
25	24.0	0.279E-05	0.286E-02	0.406E-02	0.977E-03	0.386E-08	0.369E-01	0.312E-02	0.159E-01	0.981E-05	0.106E-05	0.222E-04	0.641E-03
26	25.0	0.270E-05	0.216E-02	0.381E-02	0.709E-03	0.376E-08	0.314E-01	0.263E-02	0.159E-01	0.654E-05	0.959E-06	0.119E-04	0.538E-03
27	30.0	0.726E-06	0.548E-03	0.198E-02	0.148E-03	0.940E-09	0.142E-01	0.791E-03	0.112E-01	0.299E-05	0.163E-06	0.369E-05	0.230E-03
28	35.0	0.115E-06	0.149E-03	0.565E-03	0.334E-04	0.135E-09	0.666E-02	0.208E-03	0.429E-02	0.142E-05	0.211E-07	0.146E-06	0.103E-03
29	40.0	0.239E-07	0.428E-04	0.190E-03	0.809E-05	0.265E-10	0.324E-02	0.548E-04	0.191E-02	0.697E-06	0.330E-08	0.324E-56	0.474E-04
30	45.0	0.577E-08	0.129E-04	0.461E-04	0.206E-05	0.605E-11	0.162E-02	0.144E-04	0.607E-03	0.355E-06	0.610E-09	0.0	0.226E-04
31	50.0	0.108E-08	0.424E-05	0.119E-04	0.579E-06	0.107E-11	0.853E-03	0.381E-05	0.201E-03	0.133E-06	0.989E-10	0.0	0.114E-04
32	70.0	0.235E-11	0.509E-07	0.842E-07	0.364E-08	0.162E-14	0.713E-04	0.196E-07	0.401E-05	0.103E-07	0.477E-12	0.0	0.794E-06
33	100.0	0.150E-15	0.540E-11	0.518E-11	0.104E-12	0.604E-19	0.385E-06	0.696E-11	0.201E-08	0.553E-10	0.230E-16	0.0	0.297E-08
34	******	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0

FROM POINT: HEIGHT= 0.0 KM,N= 1,NP= 1,REF. INDEX ABOVE & BELOW X= 0.2489E-03 0.0 ,IP= 1
EQUIV. ABSORBER AMOUNTS PER KM AT X= 0.192E 01 0.879E 00 0.256E-02 0.695E 00 0.492E-01 0.910E 00 0.100E 01 0.261E-02

EQUILAVENT SEA LEVEL ABSORBER AMOUNTS

	WATER VAPOUR GM CM-2	CO2 ETC. KM	OZONE ATM CM	NITROGEN (CONT) KM	H2O (CONT) GM CM-2	MOL SCAT KM	AEROSOL KM	OZONE(UV-VIS) ATM CM
W(1-8)=	0.911E 01	0.440E 01	0.128E-01	0.348E 01	0.246E 00	0.455E 01	0.500E 01	0.131E-01

	NITRIC ACID	SO2	NO	NH3	NO2
W(11-15)=	0.0	0.990E-01	0.102E 00	0.953E-01	0.398E-01

FREQ CM-1	WAVELENGTH MICRONS	H2O TRANS	CO2+ TRANS	OZONE TRANS	N2 CONT TRANS	H2O CONT TRANS	MOL SCAT TRANS	AEROSOL TRANS	AEROSOL ABS
450	22.2222	0.0000	1.0000	1.0000	1.0000	1.0000	1.0000	0.9331	0.0376
455	21.9780	0.0000	1.0000	1.0000	1.0000	1.0000	1.0000	0.9324	0.0376

FREQ CM-1	WAVELENGTH MICRONS	SO2 TRANS	NO TRANS	NH3 TRANS	NO2 TRANS	INTEGRATED ABSORPTION	TOTAL TRANS
450	22.2222	0.9980	1.0000	1.0000	1.0000	2.5300	0.0000
455	21.9780	0.9948	1.0000	1.0000	1.0000	4.9999	0.0000

INTEGRATED ABSORPTION FROM 450 TO 455 CM-1 = 5.00,AVERAGE TRANSMITTANCE =0.0000

1

FREQUENCY RANGE V1= 550.0 CM-1 TO V2= 555.0 CM-1 FOR DV = 5.0 CM-1 (18.02 - 18.18 MICRONS)

EQUIVALENT SEA LEVEL ABSORBER AMOUNTS

	WATER VAPOUR GM CM-2	CO2 ETC. KM	OZONE ATM CM	NITROGEN (CONT) KM	H2O (CONT) GM CM-2	MOL SCAT KM	AEROSOL KM	OZONE(UV-VIS) ATM CM
W(1-9)=	0.911E 01	0.440E 01	0.128E-01	0.348E 01	0.246E 00	0.455E 01	0.500E 01	0.131E-01

	NITRIC ACID	SO2	NO	NH3	NO2
W(11-15)=	0.0	0.990E-01	0.102E 00	0.953E-01	0.398E-01

FREQ CM-1	WAVELENGTH MICRONS	H2O TRANS	CO2+ TRANS	OZONE TRANS	N2 CONT TRANS	H2O CONT TRANS	MOL SCAT TRANS	AEROSOL TRANS	AEROSOL ABS
550	18.1818	0.0211	0.9365	1.0000	1.0000	1.0000	1.0000	0.9332	0.0343
555	18.0180	0.0232	0.9190	1.0000	1.0000	1.0000	1.0000	0.9324	0.0349

FREQ CM-1	WAVELENGTH MICRONS	SO2 TRANS	NO TRANS	NH3 TRANS	NO2 TRANS	INTEGRATED ABSORPTION	TOTAL TRANS
550	18.1818	0.9175	1.0000	1.0000	1.0000	2.4577	0.0169
555	18.0180	0.9338	1.0000	1.0000	1.0000	4.9113	0.0185

INTEGRATED ABSORPTION FROM 550 TO 555 CM-1 = 4.91, AVERAGE TRANSMITTANCE =0.0177
1

FREQUENCY RANGE V1= 650.0 CM-1 TO V2= 655.0 CM-1 FOR DV = 5.0 CM-1 (15.27 - 15.38 MICRONS)

EQUILAVENT SEA LEVEL ABSORBER AMOUNTS

	WATER VAPOUR GM CM-2	CO2 ETC. KM	OZONE ATM CM	NITROGEN (CONT) KM	H2O (CONT) GM CM-2	MOL SCAT KM	AEROSOL KM	OZONE(UV-VIS) ATM CM
W(1-9)=	0.911E 01	0.440E 01	0.128E-01	0.348E 01	0.246E 00	0.455E 01	0.500E 01	0.131E-01

	NITRIC ACID	SO2	NO	NH3	NO2
W(11-15)=	0.0	0.990E-01	0.102E 00	0.953E-01	0.398E-01

FREQ CM-1	WAVELENGTH MICRONS	H2O TRANS	CO2+ TRANS	OZONE TRANS	N2 CONT TRANS	H2O CONT TRANS	MOL SCAT TRANS	AEROSOL TRANS	AEROSOL ABS
650	15.3846	0.2465	0.0000	0.9983	1.0000	1.0000	1.0000	0.9337	0.0390
655	15.2672	0.2866	0.0000	0.9975	1.0000	1.0000	1.0000	0.9338	0.0396

FREQ CM-1	WAVELENGTH MICRONS	SO2 TRANS	NO TRANS	NH3 TRANS	NO2 TRANS	INTEGRATED ABSORPTION	TOTAL TRANS
650	15.3846	1.0000	1.0000	1.0000	0.9996	2.5000	0.0000
655	15.2672	1.0000	1.0000	1.0000	0.9997	5.0000	0.0000

INTEGRATED ABSORPTION FROM 650 TO 655 CM-1 = 5.00,AVERAGE TRANSMITTANCE =0.0000

FREQUENCY RANGE V1= 750.0 CM-1 TO V2= 755.0 CM-1 FOR DV = 5.0 CM-1 (13.25 - 13.33 MICRONS)

EQUILAVENT SEA LEVEL ABSORBER AMOUNTS

	WATER VAPOUR GM CM-2	CO2 ETC. KM	OZONE ATM CM	NITROGEN (CONT) KM	H2O (CONT) GM CM-2	MOL SCAT KM	AEROSOL KM	OZONE(UV-VIS) ATM CM
W(1-8)=	0.911E 01	0.440E 01	0.128E-01	0.348E 01	0.246E 00	0.455E 01	0.500E 01	0.131E-01

	NITRIC ACID	SO2	NO	NH3	NO2
W(11-15)=	0.0	0.930E-01	0.102E 00	0.953E-01	0.399E-01

```
 FREQ WAVELENGTH   H2O     CO2+    OZONE   N2 CONT  H2O CONT MOL SCAT  AEROSOL  AEROSOL
 CM-1  MICRONS    TRANS    TRANS   TRANS    TRANS    TRANS    TRANS    TRANS     ABS
  950  10.5263   0.8800   0.9603  0.9994   1.0000   0.1642   1.0000   0.9231   0.0341
  955  10.4712   0.9681   0.9612  0.9988   1.0000   0.1692   1.0000   0.9226   0.0347

 FREQ WAVELENGTH   SO2      NO      NH3      NO2  INTEGRATED  TOTAL
 CM-1  MICRONS    TRANS    TRANS   TRANS    TRANS ABSORPTION  TRANS
  950  10.5263   1.0000   1.0000  0.9212   1.0000   2.2052   0.1179
  955  10.4712   1.0000   1.0000  0.8878   1.0000   4.4165   0.1155
INTEGRATED ABSORPTION FROM  950 TO  955 CM-1 =     4.42,AVERAGE TRANSMITTANCE =0.1167
 1

 FREQUENCY RANGE V1=  1150.0 CM-1 TO V2=  1155.0 CM-1 FOR DV =   5.0 CM-1   (  8.66 -  8.70 MICRONS )

 EQUILAVENT SEA LEVEL ABSORBER AMOUNTS

             WATER VAPOUR    CO2 ETC.     OZONE    NITROGEN (CONT)  H2O (CONT)    MOL  SCAT     AEROSOL  OZONE(UV-VIS)
               GM CM-2          KM        ATM CM         KM          GM CM-2         KM            KM         ATM CM

 W(1-9)=      0.911E 01    0.440E 01    0.128E-01    0.348E 01     0.246E 00     0.455E 01    0.500E 01    0.131E-01


             NITRIC ACID       SO2          NO          NH3          NO2

 W(11-15)=      0.0        0.990E-01    0.102E 00    0.953E-01    0.398E-01
```

FREQ CM-1	WAVELENGTH MICRONS	H2O TRANS	CO2+ TRANS	OZONE TRANS	N2 CONT TRANS	H2O CONT TRANS	MOL SCAT TRANS	AEROSOL TRANS	AEROSOL ABS
750	13.3333	0.4889	0.4739	0.9980	1.0000	0.0084	1.0000	0.9361	0.0261
755	13.2450	0.5450	0.5962	0.9983	1.0000	0.0097	1.0000	0.9359	0.0260

FREQ CM-1	WAVELENGTH MICRONS	SO2 TRANS	NO TRANS	NH3 TRANS	NO2 TRANS	INTEGRATED ABSORPTION	TOTAL TRANS
750	13.3333	1.0000	1.0000	0.9814	0.9994	2.4956	0.0018
755	13.2450	1.0000	1.0000	0.9783	0.9995	4.9884	0.0029

INTEGRATED ABSORPTION FROM 750 TO 755 CM-1 = 4.99,AVERAGE TRANSMITTANCE =0.0023
1

FREQUENCY RANGE V1= 950.0 CM-1 TO V2= 955.0 CM-1 FOR DV = 5.0 CM-1 (10.47 - 10.53 MICRONS)

EQUILAVENT SEA LEVEL ABSORBER AMOUNTS

	WATER VAPOUR GM CM-2	CO2 ETC. KM	OZONE ATM CM	NITROGEN (CONT) KM	H2O (CONT) GM CM-2	MOL SCAT KM	AEROSOL KM	OZONE(UV-VIS) ATM CM
W(1-8)=	0.911E 01	0.440E 01	0.129E-01	0.348E 01	0.246E 00	0.455E 01	0.500E 01	0.131E-01

	NITRIC ACID	SO2	NO	NH3	NO2
W(11-15)=	0.0	0.990E-01	0.102E 00	0.953E-01	0.398E-01

FREQ CM-1	WAVELENGTH MICRONS	H2O TRANS	CO2+ TRANS	OZONE TRANS	N2 CONT TRANS	H2O CONT TRANS	MOL SCAT TRANS	AEROSOL TRANS	AEROSOL ABS
1150	8.6957	0.7275	0.9622	0.9973	1.0000	0.3041	1.0000	0.9055	0.0548
1155	8.6580	0.7094	0.9612	0.9980	1.0000	0.3060	1.0000	0.9373	0.0560

FREQ CM-1	WAVELENGTH MICRONS	SO2 TRANS	NO TRANS	NH3 TRANS	NO2 TRANS	INTEGRATED ABSORPTION	TOTAL TRANS
1150	8.6957	0.9040	1.0000	0.9788	1.0000	2.0748	0.1701
1155	8.6580	0.8929	1.0000	0.9820	1.0000	4.1606	0.1657

INTEGRATED ASORPTION FROM 1150 TO 1155 CM-1 = 4.16,AVERAGE TRANSMITTANCE =0.1679
1

FREQUENCY RANGE V1= 1350.0 CM-1 TO V2= 1355.0 CM-1 FOR DV = 5.0 CM-1 (7.38 - 7.41 MICRONS)

EQUILAVENT SEA LEVEL ABSORBER AMOUNTS

	WATER VAPOUR GM CM-2	CO2 ETC. KM	OZONE ATM CM	NITROGEN (CONT) KM	H2O (CONT) GM CM-2	MOL SCAT KM	AEROSOL KM	OZONE(UV-VIS) ATM CM
W(1-8)=	0.911E 01	0.440E 01	0.128E-01	0.348E 01	0.246E 00	0.455E 01	0.500E 01	0.131E-01

	NITRIC ACID	SO2	NO	NH3	NO2
W(11-15)=	0.0	0.590E-01	0.102E 00	0.953E-01	0.398E-01

FREQ CM-1	WAVELENGTH MICRONS	H2O TRANS	CO2+ TRANS	OZONE TRANS	N2 CONT TRANS	H2O CONT TRANS	MOL SCAT TRANS	AEROSOL TRANS	AEROSOL ABS
1350	7.4074	0.0001	0.6340	1.0000	1.0000	0.3456	1.0000	0.9310	0.0347
1355	7.3801	0.0001	0.6823	1.0000	1.0000	1.0000	1.0000	0.9302	0.0350

FREQ CM-1	WAVELENGTH MICRONS	SO2 TRANS	NO TRANS	NH3 TRANS	NO2 TRANS	INTEGRATED ABSORPTION	TOTAL TRANS
1350	7.4074	0.2780	1.0000	1.0000	1.0000	2.5000	0.0000
1355	7.3801	0.2728	1.0000	1.0000	1.0000	5.0000	0.0000

INTEGRATED ASORPTION FROM 1350 TO 1355 CM-1 = 5.00,AVERAGE TRANSMITTANCE =0.0000
1

FREQUENCY RANGE V1= 1850.0 CM-1 TO V2= 1855.0 CM-1 FOR DV = 5.0 CM-1 (5.39 - 5.41 MICRONS)

EQUILAVENT SEA LEVEL ABSORBER AMOUNTS

	WATER VAPOUR GM CM-2	CO2 ETC. KM	OZONE ATM CM	NITROGEN (CONT) KM	H2O (CONT) GM CM-2	MOL SCAT KM	AEROSOL KM	OZONE(UV-VIS) ATM CM
W(1-8)=	0.911E 01	0.440E 01	0.128E-01	0.348E 01	0.246E 00	0.455E 01	0.500E 01	0.131E-01

	NITRIC ACID	SO2	NO	NH3	NO2
W(11-15)=	0.0	0.590E-01	0.102E 00	0.953E-01	0.398E-01

FREQ CM-1	WAVELENGTH MICRONS	H2O TRANS	CO2+ TRANS	OZONE TRANS	N2 CONT TRANS	H2O CONT TRANS	MOL SCAT TRANS	AEROSOL TRANS	AEROSOL ABS
2450	4.0816	0.9922	0.9617	1.0000	0.7665	0.8288	1.0000	0.9212	0.0119
2455	4.0733	0.9914	0.9645	1.0000	0.7827	0.8320	1.0000	0.9211	0.0119

FREQ CM-1	WAVELENGTH MICRONS	SO2 TRANS	NO TRANS	NH3 TRANS	NO2 TRANS	INTEGRATED ABSORPTION	TOTAL TRANS
2450	4.0816	1.0000	1.0000	1.0000	1.0000	1.1041	0.5584
2455	4.0733	0.9998	1.0000	1.0000	1.0000	2.1705	0.5734

INTEGRATED ASORPTION FROM 2450 TO 2455 CM-1 = 2.17,AVERAGE TRANSMITTANCE =0.5659
1

FREQUENCY RANGE V1= 3150.0 CM-1 TO V2= 3155.0 CM-1 FOR DV = 5.0 CM-1 (3.17 - 3.17 MICRONS)

EQUILAVENT SEA LEVEL ABSORBER AMOUNTS

	WATER VAPOUR GM CM-2	CO2 ETC. KM	OZONE ATM CM	NITROGEN (CONT) KM	H2O (CONT) GM CM-2	MOL SCAT KM	AEROSOL KM	OZONE(UV-VIS) ATM CM
W(1-8)=	0.911E 01	0.440E 01	0.128E-01	0.348E 01	0.246E 00	0.455E 01	0.500E 01	0.131E-01

	NITRIC ACID	SO2	NO	NH3	NO2
W(11-15)=	0.0	0.990E-01	0.102E 00	0.953E-01	0.398E-01

FREQ CM-1	WAVELENGTH MICRONS	H2O TRANS	CO2+ TRANS	OZONE TRANS	N2 CONT TRANS	H2O CONT TRANS	MOL SCAT TRANS	AEROSOL TRANS	AEROSOL ABS
1850	5.4054	0.0000	0.9955	0.9999	1.0000	1.0000	1.0000	0.9283	0.0186
1855	5.3908	0.0000	0.9949	0.9999	1.0000	1.0000	1.0000	0.9282	0.0185

FREQ CM-1	WAVELENGTH MICRONS	SO2 TRANS	NO TRANS	NH3 TRANS	NO2 TRANS	INTEGRATED ABSORPTION	TOTAL TRANS
1850	5.4054	1.0000	0.8961	1.0000	1.0000	2.5000	0.0000
1855	5.3908	1.0000	0.9034	1.0000	1.0000	5.0000	0.0000

INTEGRATED ABSORPTION FROM 1850 TO 1855 CM-1 = 5.00,AVERAGE TRANSMITTANCE =0.0000
1

FREQUENCY RANGE V1= 2450.0 CM-1 TO V2= 2455.0 CM-1 FOR DV = 5.0 CM-1 (4.07 - 4.06 MICRONS)

EQUILAVENT SEA LEVEL ABSORBER AMOUNTS

	WATER VAPOUR GM CM-2	CO2 ETC. KM	OZONE ATM CM	NITROGEN (CONT) KM	H2O (CONT) GM CM-2	MOL SCAT KM	AEROSOL KM	OZONE(UV-VIS) ATM CM
W(1-8)=	0.911E 01	0.440E 01	0.128E-01	0.348E 01	0.246E 00	0.455E 01	0.500E 01	0.131E-01

	NITRIC ACID	SO2	NO	NH3	NO2
W(11-15)=	0.0	0.590E-01	0.102E 00	0.953E-01	0.398E-01

FREQ CM-1	WAVELENGTH MICRONS	H2O TRANS	CO2+ TRANS	OZONE TRANS	N2 CONT TRANS	H2O CONT TRANS	MOL SCAT TRANS	AEROSOL TRANS	AEROSOL ABS
3150	3.1746	0.4800	0.9190	0.9998	1.0000	1.0000	1.0000	0.9116	0.0231
3155	3.1696	0.5021	0.9373	0.9998	1.0000	1.0000	1.0000	0.9116	0.0233

FREQ CM-1	WAVELENGTH MICRONS	SO2 TRANS	NO TRANS	NH3 TRANS	NO2 TRANS	INTEGRATED ABSORPTION	TOTAL TRANS
3150	3.1746	1.0000	1.0000	1.0000	1.0000	1.4951	0.4020
3155	3.1696	1.0000	1.0000	1.0000	1.0000	2.9229	0.4289

INTEGRATED ASORPTION FROM 3150 TO 3155 CM-1 = 2.92,AVERAGE TRANSMITTANCE =0.4154

0

```
C
C        ADSET   V 1-3
C
C        EVALUATION OF ABSORBER PARAMETERS AND STANDARD EMPIRICAL AND
C        PIECEWISE-ANALYTICAL TRANSMISSION FUNCTIONS
C
C        THIS CODE USES THE SUBROUTINE SIMQ IN SSP LIBRARY
C
C     THIS CODE CONSISTS OF
C        MAIN:  COMPUTATIONS OF BAND PARAMETERS N,M,C'
C               AND EMPIRICAL STANDARD TRANSMISSION FUNCTION
C        NMBC:  COMPUTATION OF NON-MAJOR BANDS' C-VALUES
C        INTPL1:  COMPUTATION OF THE PIECEWISE-ANALYTICAL
C                 STANDARD TRANSMISSION FUNCTION GIVEN BY
C                 TAU = EXP(-10**(A1+A2*X))
C        INTPL2:  COMPUTATION OF THE PIECEWISE-ANALYTICAL
C                 STANDARD TRANSMISSION FUNCTION GIVEN BY
C                 TAU = EXP(-10**(A1+A2*X+A3*X**2))
C        SDTAU:  COMPUTATION OF THE ERROR STANDARD DERIVATIONS
C                BETWEEN PIECEWISE ANALYTICAL STANDARD
C                TRANSMISSION FUNCTION AND THE ORIGINAL DATA
C                USED IN THE MAIN PROGRAMME
C
C      DATA SET-UP
C         1. 1ST CARD   TITLE IN 20A4
C                      (ABSORBER TYPE,ETC)
C         2. 2ND CARD   FOUR CONTROL NUMBERS IN 4I5
C                MAXRPT  MAXIMUN NUMBER OF REPETITION OF
C                        THE COMPUTATION IN MAIN
C                INDX(1)  SUBROUTINE NMBC IS CALLED IF > 0
C                INDX(2)  SUBROUTINE INTPL1 IS CALLED IF > 0
C                INDX(3)  SUBROUTINE INTPL2 IS CALLED IF > 0
C         3. DATA SET FOR MAIN
C                 CONSISTS OF SEVERAL SUBSETS (MAX.6) OF DATA
C                 CORRESPONDING TO INDIVIDUAL MAJOR BANDS
C                 FIRST CARD FOR EACH SUBSET IS A CONTROL CARD WHICH
C                 CONTAINS BAND#, WAVE NUMBER, # OF CUTS AND # OF LEVELS
C                 IN THIS ORDER BY THE FORMAT (I5,F10.3,2I5)
C                 REFER TO THE READ(5,105) AND FORMAT 105 FOR THE
C                 CONTENTS OF EACH CARD.
C                 END OF DATA IS DEFINED BY A BLANK CONTROL CARD
C          4. DATA SET FOR NMBC
C                  INDIVIDUAL DATA FORMAT - SAME AS FOR MAIN
C                  END OF EACH DATA SET FOR A BAND IS MARKED BY
C                  A BLANK CARD
C                  END OF ALL DATA IS MARKED BY -1 (I2) IN ADDITION TO
C                  A BLANK CARD
C                  IF NO DATA BUT A BLANK CARD IS SUPPLIED, THEN
C                  THIS SUBROUTINE IS SKIPPED.
C          5. DATA SET FOR INTPL1
C                  FORMATION OF THE DATA IS THE SAME AS THAT  FOR MAIN
C                  DATA (IF SUPPLIED) WILL BE USED FOR S.D. COMPUTATION
C                  ONLY.
C                  IF NO DATA BUT A BLANK CARD IS SUPPLIED, THEN THE
C                  S.D. COMPUTATION IS SKIPPED.
C          6. DATA SET FOR INTPL2
```

```
C                    FORMATION OF THE DATA IS THE SAME AS THAT FOR MAIN
C                    IF NO DATA BUT A BLANK CARD IS SUPPLIED, THEN
C                    THIS SUBROUTINE IS SKIPPED.
C     NOTE:   DATA SET MUST HAVE A CUT STRUCTURE SUCH THAT EQUAL
C             TRANSMITTANCE DATA ARE GROUPED TOGETHER AND THESE GROUPS
C             ARE QUEUED IN THE DECENDING ORDER IN TAU. THE QUEUING OF
C             THE LEVELS WITHIN EVERY GROUP MUST BE THE SAME.
C
      DIMENSION V(19),A(19,19),X(361),B(19),RI(6,12,10)
      DIMENSION P(10),WWW(12,10),STANDV(12),TSD(6),NDATA(6),INDX(3)
      COMMON /PARM1/ TSTD(12),PW(12),WN(6),CSTD(6),NCUT,NC,NAME(20),
     *               AN,AM,CF,ICONST(6),NEL
      COMMON /PARM2/ PRES(6,12,10),TEMP(6,12,10),UGAS(6,12,10),
     *               TAU(6,12),NTC(6),NLV(6)
      CF=1.0
      LOOPCT=1
      WCRIT=2.
      M=0
      READ(5,100) (NAME(I),I=1,20)
  100 FORMAT(20A4)
      READ(5,101) MAXRPT,(INDX(I),I=1,3)
  101 FORMAT(4I5)
C
C        COMPUTATION OF ABSORBER PARAMETERS N, M & C-VALUES IS REPEATED
C         MAXRPT TIMES, WHERE 1 < MAXRPT < 10 IS READ IN BY I5 FORMAT
C         (SUGGESTED VALUE IS 5)
C
C        DATA READ-IN ROUTINE
C
 1000 CONTINUE
      READ(5,102) IC,W,JM,KM
  102 FORMAT(I5,F10.3,2I5)
      IF(IC.LE.0) GO TO 2000
      IF(M.GT.0) GO TO 10
      CALL DATE (MONTH,IDAY,IYEAR)
      WRITE(6,111)MONTH,IDAY,IYEAR
  111 FORMAT(1H1,T60,I4,' / ',I2,' / ',I2,//)
      WRITE(6,200) (NAME(I),I=1,20)
  200 FORMAT(1H ,T25,20A4)
      GO TO 11
   10 CONTINUE
      WRITE(6,201)
  201 FORMAT(1H1)
   11 CONTINUE
      M=M+1
      WN(M)=W
      NTC(M)=JM
      NLV(M)=KM
      WRITE(6,202) M,WN(M),NTC(M),NLV(M)
  202 FORMAT(1H0,T15,'***  BAND',I3,'  (WAVE NUMBER =',F10.3,')  ***',
     * ///,T20,'TOTAL # OF CUTS   =',I3,//,T20,'TOTAL # OF LEVELS =',I3,
     * ///)
      WRITE(6,203)
  203 FORMAT(1H ,T5,'( DATA FORMAT )',//,T9,'GAS#',T15,'WAVE #',T24,
     * 'PRESSURE',T36,'TEMP.',T46,'PPM',T58,'RANGE',T70,'UGAS',T78,
     * 'TRANSM.',/)
```

```
      DO 12 J=1,JM
      WRITE(6,204) J
  204 FORMAT(1H0,T5,'< CUT',I3,' >',/)
      T=0.
      IT=0
      DO 13 K=1,KM
      READ(5,103)  KGAS,FREQ,RPRES,RTEMP,PPM,RANGE,RUGAS,TX
  103 FORMAT(I2,F10.3,E11.4,F9.3,E11.4,E13.6,E11.4,F7.4)
      RUGAS=RUGAS/CF
      WRITE(6,205) KGAS,FREQ,RPRES,RTEMP,PPM,RANGE,RUGAS,TX
  205 FORMAT(T10,I2,F10.3,E11.4,F9.3,E11.4,E13.6,E11.4,F8.4)
C
C         PRES, TEMP & UGAS ARE CONVERTED TO THE LOG OF THE NORMALIZED
C         VALUES. IF RPRES=0 (INDICATES NO DATA),THEN UGAS(M,J,K) IS SET
C         AT AN IMPOSSIBLE VALUE , ALSO RI(M,J,K) IS SET TO ZERO.
C
      IF(RPRES.GT.0.) GO TO 14
      PRES(M,J,K)=0.
      TEMP(M,J,K)=0.
      UGAS(M,J,K)=10.
      RI(M,J,K)=0.
      GO TO 15
   14 CONTINUE
      PRES(M,J,K)=ALOG10(RPRES/1013.)
      TEMP(M,J,K)=ALOG10(273.15/RTEMP)
      UGAS(M,J,K)=ALOG10(RUGAS)
   15 CONTINUE
C
      T=T+TX
      IT=IT+1
      RI(M,J,K)=1.0
   13 CONTINUE
      TAU(M,J)=T/FLOAT(IT)
   12 CONTINUE
      GO TO 1000
C
C         END OF DATA INPUT
C
C         CONSTANTS USED IN LATER COMPUTATION ARE INITIALIZED
C         FROM 2000 TO 3000.
C         NCUT = MAXIMUM # OF CUTS USED IN COMPUTATION
C         NDIM = DIMENSION OF THE COEFFICIENT MATRIX
C
 2000 CONTINUE
      IF(M.GT.0) GO TO 20
      WRITE(6,206)
  206 FORMAT(1H0,//,T10,'$$$  NO INPUT DATA  $$$')
      STOP
   20 CONTINUE
      NC=M
      CSTD(1)=0.
      NCUT=NTC(1)
      NPTS=NTC(1)*NLV(1)
      IF(NC.LE.1) GO TO 21
      DO 22 I=2,NC
      NCUT=MAX0(NCUT,NTC(I))
```

```
      NPTS=NPTS+NTC(I)*NLV(I)
   22 CONTINUE
   21 CONTINUE
      FNC=FLOAT(NC)
      RIT=FLOAT(NPTS)
      DO 23 J=1,NCUT
      TC=0.
      DO 24 M=1,NC
      TC=TC+TAU(M,J)
   24 CONTINUE
      TSTD(J)=TC/FNC
   23 CONTINUE
      NDIM=NC+1+NCUT
C
C*************************************************************************
C        COMPUTATION OF THE ABSORBER PARAMETERS.
C        THIS LOOP WILL BE REPEATED MAXRPT TIMES.
C
C        FORMATION OF THE NORMAL EQUATION AX = B , A IS SYMMETRIC
C
 3000 CONTINUE
      DO 30 I=1,19
      B(I)=0.
      DO 31 J=1,19
      A(I,J)=0.
   31 CONTINUE
   30 CONTINUE
C
      DO 1 M=1,NC
      JM=NTC(M)
      KM=NLV(M)
      DO 2 J=1,JM
      DO 3 K=1,KM
      IF(RI(M,J,K).LT.0.5.) GO TO 3
      DO 4 IC=1,19
      V(IC)=0.
    4 CONTINUE
      V(NDIM-M)=1.
      V(NDIM)=PRES(M,J,K)
      V(NDIM-1)=TEMP(M,J,K)
      VV=-UGAS(M,J,K)
      V(NCUT+1-J)=1.
      DO 5 II=1,NDIM
      DO 6 IJ=1,NDIM
      A(II,IJ)=V(II)*V(IJ)*RI(M,J,K) + A(II,IJ)
    6 CONTINUE
      B(II)=V(II)*VV*RI(M,J,K) + B(II)
    5 CONTINUE
    3 CONTINUE
    2 CONTINUE
    1 CONTINUE
C
C        IF J-TH ROW OF "A" IS ZERO, A(J,J) IS CHANGED TO -1
C         WHICH IS DONE IN ORDER TO MAKE "A" NON-SINGULAR
C         THIS HAPPENS WHEN ALL OF THE DATA FOR BAND J-1 FAIL TO SATISFY
C         THE CRITERION W < WCRIT.  THE BAND J-1 WILL BE IGNORED IF THIS
```

```
         HAPPENS, AND THE C-VALUE FOR BAND J-1 WILL BE COMPUTED
C        SEPARETELY.
C
      ICONST(1)=1
      IF(NC.EQ.1) GO TO 40
      DO 41 M=2,NC
      ICONST(M)=1
      I=NDIM-M
      IF(A(I,I).NE.0.) GO TO 41
      A(I,I)=-1.0
      ICONST(M)=0
   41 CONTINUE
   40 CONTINUE
C
C
      NCOL=0
      DO 42 J=1,NDIM
      DO 43 I=1,NDIM
      NCOL=NCOL+1
      X(NCOL)=A(I,J)
   43 CONTINUE
   42 CONTINUE
C
C        PRINTING OF THE HEADING FOR EACH TRIAL AND THE NORMAL EQUATION
C
      IF(LOOPCT.GT.1) GO TO 50
      WRITE(6,207) MAXRPT,LOOPCT
  207 FORMAT(1H1,T20,'***  ABSORBER PARAMETER COMPUTATION  ***',///,
     * T15,'NOTE:  THE COMPUTATION WILL BE REPEATED MAXRPT =',I2,
     * ' TIMES.',////,T10,'TRIAL #',I1,5X,'(ALL DATA WERE USED)')
      GO TO 51
   50 CONTINUE
      WRITE(6,208) LOOPCT
  208 FORMAT(1H1,T10,'TRIAL #',I1,5X,'(PARTIAL DATA WERE USED WITH',
     * ' CUT-OFF CRITERION :  W < 2 )')
   51 CONTINUE
      WRITE(6,209) NDIM,NDIM
  209 FORMAT(//,1H0,'< NORMAL EQUATION : AX = B >',//,T10,',WHERE THE',
     * ' COEFFICIENT MATRIX A(',I3,' ,',I3,' ) AND THE CONSTANT VECTOR',
     * ' B ARE',//)
      IF(NDIM.LE.17) GO TO 52
      WRITE(6,210) NDIM
  210 FORMAT(1H ,'***  WARNING : DIMENSION OF THE MATRIX IS TOO LARGE',
     * ' (',I3,' ) TO BE PRINTED IN A MATRIX FORM  ***',/)
   52 CONTINUE
      DO 53 I=1,NDIM
      WRITE(6,211) (A(I,J),J=1,NDIM),B(I)
  211 FORMAT(1H ,18F7.3)
   53 CONTINUE
C
C*****   MATRIX INVERSION SUBROUTINE SIMQ IN SSP IS CALLED **************
C
      CALL SIMQ(X,B,NDIM,KS)
C
C        PRINTING OF THE SOLUTION FOR THE NORMAL EQUATION
C
```

```
      IF(KS.EQ.1) WRITE(6,212)
  212 FORMAT(1H0,T10,'WARNING:  THE COEFFICIENT MATRIX IS SINGULAR.')
      AN=B(NDIM)
      AM=B(NDIM-1)
      IF(NC.LE.1) GO TO 54
      DO 55 M=2,NC
      CSTD(M)=B(NDIM-M)
   55 CONTINUE
   54 CONTINUE
      DO 56 J=1,NCUT
      PW(J)=-B(NCUT+1-J)
   56 CONTINUE
      WRITE(6,213) AN,AM,(CSTD(M),M=1,NC)
  213 FORMAT(//,1H0,' < RESULTS >',///,T7,'N',T17,'M',T27,'C1',T37,'C2',
     * T47,'C3',T57,'C4',T67,'C5',T77,'C6',//,2F10.5,6F10.3)
      WRITE(6,214) (PW(I),I=1,NCUT)
  214 FORMAT(/,1H0,T7,'X*1',T17,'X*2',T27,'X*3',T37,'X*4',T47,'X*5',
     * T57,'X*6',T67,'X*7',T77,'X*8',T87,'X*9',T97,'X*10',T107,'X*11',
     * T117,'X*12',//,12F10.3)
      NEL=NPTS-INT(RIT)
      WRITE(6,215) NEL
  215 FORMAT(//,1H0,T4,'# OF ELIMINATED POINTS  =',I5)
C
C        CHECKING OF THE CRITERION ( W < WCRIT ) AND THE COMPUTATION
C        OF C-VALUES FOR THE IGNORED BANDS.
C        RI(M,J,K) = 0 IF  W  IS GREATER THAN OR EQUAL TO WCRIT
C        RI(M,J,K) = 1 IF  W  IS LESS THAN WCRIT
C
      RIT=0.
      DO 60 M=1,NC
      JM=NTC(M)
      KM=NLV(M)
      CAVG=0.0
      DO 61 J=1,JM
      DO 62 K=1,KM
      W=AN*PRES(M,J,K)+AM*TEMP(M,J,K)+UGAS(M,J,K)
      IF(W.GE.WCRIT) RI(M,J,K)=0.
      RIT=RIT+RI(M,J,K)
      CAVG=CAVG+(PW(J)-W)
   62 CONTINUE
   61 CONTINUE
      IF(ICONST(M).EQ.1) GO TO 60
      CSTD(M)=CAVG/FLOAT(JM*KM)
      WRITE(6,216) M,M,M,CSTD(M)
  216 FORMAT(///,1H ,T7,'** WARNING **',T25,'NO DATA FOR BAND',I2,
     * ' SATISFIES THE CRITERION ( W < 2 ).',//,T25,'THE C',I1,
     * ' VALUE IS SEPARATELY COMPUTED BY AVERAGING.',//,T30,'C',I1,
     * ' =',F10.3)
   60 CONTINUE
C
C        COMPUTATIONS OF STANDARD DEVIATIONS IN X
C
      NGDATA=0
      GTSD=0.
      ICST=NC
      DO 70 M=1,NC
```

```
      JM=NTC(M)
      KM=NLV(M)
      NDATA(M)=0
      TSD(M)=0.
      WRITE(6,201)
      WRITE(6,202) M,WN(M),NTC(M),NLV(M)
      WRITE(6,217) AN,AM,M,CSTD(M)
  217 FORMAT(1H ,T10,'N  =',F10.5,//,T10,'M  =',F10.5,//,T10,'C',I1,
     * ' =',F10.5)
      WRITE(6,218)
  218 FORMAT(//,1H0,T7,'RECOMPUTED X-VALUES AND STANDARD DEVIATIONS',
     * ' IN X-VALUES',/,1H0,T2,'CUT',T11,'TAU',T20,'X*',T30,'X1',T39,
     * 'X2',T48,'X3',T57,'X4',T66,'X5',T75,'X6',T84,'X7',T93,'X8',
     * T102,'X9',T111,'X10',T121,'CUTWISE-SD',/)
C
C        COMPUTATION OF THE CUTWISE STANDARD DEVIATIONS IN X
C
      DO 71 J=1,JM
      DN=0.
      WW=0.
      DO 72 K=1,KM
      P(K)=CSTD(M)+AN*PRES(M,J,K)+AM*TEMP(M,J,K)+UGAS(M,J,K)
      WWW(J,K)=(PW(J)-P(K))**2*RI(M,J,K)
      WW=WW+WWW(J,K)
      DN=DN+RI(M,J,K)
   72 CONTINUE
      WW=SQRT(WW/DN)
      NDATA(M)=NDATA(M)+IFIX(DN)
      WRITE(6,219) J,TAU(M,J),PW(J),(P(K),K=1,KM)
  219 FORMAT(1H ,I5,F9.3,F9.4,1X,10F9.4)
      WRITE(6,220) WW
  220 FORMAT(1H+,T121,F10.5)
   71 CONTINUE
C
C        COMPUTATION OF THE LEVELWISE STANDARD DEVIATIONS IN X
C
      DO 73 K=1,KM
      WW=0.
      DN=0.
      DO 74 J=1,JM
      WW=WW+WWW(J,K)
      DN=DN+RI(M,J,K)
   74 CONTINUE
      TSD(M)=TSD(M)+WW
      STANDV(K)=SQRT(WW/DN)
   73 CONTINUE
      WRITE(6,221) (STANDV(K),K=1,KM)
  221 FORMAT(1H0,T4,'LEVELWISE-SD :',T26,10F9.5)
      NGDATA=NGDATA+NDATA(M)*ICONST(M)
      GTSD=GTSD+TSD(M)*FLOAT(ICONST(M))
      ICST=ICST-ICONST(M)
      TSD(M)=SQRT(TSD(M)/FLOAT(NDATA(M)))
      WRITE(6,222) TSD(M)
  222 FORMAT(//,1H0,T4,'TOTAL STANDARD DEVIATION FOR THIS BAND  :',
     * F15.6)
   70 CONTINUE
```

```
C
C
C        PRINTOUT OF THE SUMMARY.
C        ALL VITAL INFORMATIONS ARE PRINTED OUT HERE.
C
C
      GTSD=SQRT(GTSD/FLOAT(NGDATA))
      WRITE(6,223) LOOPCT,AN,AM
  223 FORMAT(1H1,T15,'***  SUMMARY OF THE ABSORBER PARAMETER',
     * ' COMPUTATION FOR TRIAL #',I2,'  ***',///,T20,
     * 'PRESSURE EXPONENT    N =',F10.5,//,T20,
     * 'TEMPERATURE EXPONENT M =',F10.5,///,T5,'CASE #',3X,
     * 'WAVE NUMBER',5X,'C-VALUE',5X,'TOTAL # OF DATA',3X,
     * 'CASEWISE S.D. IN P')
      WRITE(6,224) (M,WN(M),CSTD(M),NDATA(M),TSD(M),M=1,NC)
  224 FORMAT(1H0,T6,I3,6X,F9.2,5X,F8.3,10X,I3,12X,F12.6)
      WRITE(6,225) NGDATA,NEL,GTSD
  225 FORMAT(//,1H0,T15,'GRAND TOTAL # OF DATA =',I5,//,T15,'# OF',
     * ' ELIMINATED DATA =',I5,//,T15,'GLOBAL STANDARD DEVIATION IN P',
     * ' =',F12.6,//)
      IF(ICST.LE.0) GO TO 75
      DO 76 M=1,NC
      IF(ICONST(M).EQ.1) GO TO 76
      WRITE(6,226) M
  226 FORMAT(1H ,T15,'NOTE:  THE BAND',I3,' IS NOT INCLUDED IN THE',
     * ' FINAL STANDARD DEVIATION')
   76 CONTINUE
   75 CONTINUE
      WRITE(6,227) LOOPCT,(TSTD(J),PW(J),J=1,NCUT)
  227 FORMAT(///,1H0,T15,'***  STANDARD EMPIRICAL TRANSMISSION',
     * ' FUNCTION FOR TRIAL #',I2,'  ***',///,T20,'TAU',T35,'X*',/,
     * (1H0,T17,F7.3,T30,F8.4))
C
C
      IF(RIT.GT.0.) GO TO 80
C
C        IF NO INPUT DATA SATISFIES THE CRITERION, THE COMPUTATION IS
C        TERMINATED. THE MOST RECENT RESULTS WILL BE USED IN THE SEQUAL.
C
      WRITE(6,228)
  228 FORMAT(1H1,//,T15,'$$$  NO INPUT DATA SATISFIES THE CRITERION OF',
     * ' ( W < 2 )  $$$',//,T15,'$$$  THE COMPUTATION FOR THIS STEP IS',
     * ' TERMINATED  $$$')
      GO TO 4000
   80 CONTINUE
      LOOPCT=LOOPCT+1
      IF(LOOPCT.GT.MAXRPT) GO TO 4000
      GO TO 3000
C
 4000 CONTINUE
C
C        SUBROUTINE COMPUTATIONS FOLLOW
C
      IF(INDX(1).LE.0) GO TO 90
C
      CALL NMBC
```

```
C
   90 CONTINUE
      IF(INDX(2).LE.0) GO TO 91
C
      CALL INTPL1
C
   91 CONTINUE
      IF(INDX(3).LE.0) GO TO 92
C
      CALL INTPL2
C
   92 CONTINUE
      STOP
      END
      SUBROUTINE NMBC
C
C        COMPUTATION OF C'-VALUES FOR NON-MAJOR BANDS
C
      DIMENSION B(15),CS(15),FS(15)
      COMMON /PARM1/ TSTD(12),PW(12),WN(6),CSTD(6),NCUT,NC,NAME(20),
     *               AN,AM,CF,ICONST(6),NEL
      WRITE(6,5) (NAME(I),I=1,20)
    5 FORMAT(1H1,T15,20A4)
      WRITE(6,10)
   10 FORMAT(1H0,T15,' *** CALCULATION OF THE SPECTRAL PARAMETERS',
     * 'FOR NON-MAJOR BANDS ***'///)
      DF=1.E30
   11 CONTINUE
      NFREQ=0
   12 CONTINUE
      C=0.
      I=0
   15 CONTINUE
      READ(5,20) KGAS,FREQ,P,T,UGAS,TX
   20 FORMAT(I2,F10.3,E11.4,F9.3,24X,E11.4,F7.4)
      IF(KGAS.EQ.0) GO TO 25
      IF (KGAS.LT.0) GO TO 35
C
C        THE FOLLOWING IF-STATEMENT IS INSERTED TO DETECT
C        AND TO IGNORE THE INVALID DATA POINTS.
C
      IF(UGAS.GE.DF) GO TO 15
      I=I+1
      WX=FREQ
      UGAS=UGAS/CF
      C=C+(PW(I)-AN*ALOG10(P/1013.)-AM*ALOG10(273.15/T)-ALOG10(UGAS))
      GO TO 15
   25 C=C/FLOAT(I)
      NFREQ=NFREQ+1
      CS(NFREQ)=C
      FS(NFREQ)=WX
      DO 27 M=1,NC
      IF(ABS(WX-WN(M)).LE.0.1) CS(NFREQ)=CSTD(M)
   27 CONTINUE
      IF(NFREQ.EQ.10) GO TO 30
      GO TO 12
```

```
   30 CONTINUE
      WRITE(6,31) (FS(K),K=1,NFREQ)
   31 FORMAT(1H0,2X,'WAVE NUMBER',2X,10F11.0)
      WRITE(6,32) (CS(K),K=1,NFREQ)
   32 FORMAT(1H0,5X,'C VALUES',2X,10F11.3//)
      GO TO 11
   35 CONTINUE
      IF(NFREQ.EQ.0) GO TO 40
      WRITE(6,31) (FS(K),K=1,NFREQ)
      WRITE(6,32) (CS(K),K=1,NFREQ)
   40 CONTINUE
      RETURN
      END
C
      SUBROUTINE INTPL1
C
C        COMPUTATION OF THE STANDARD PIECEWISE-ANALYTICAL TRANSMISSION
C        FUNCTION
C
C        VERSION 1 - 1   ** A3(I) = 0 **
C        TAU = EXP(-10**( A1(I)+A2(I)*X ))
C
      DIMENSION SDCUT(15),ICUT(15),SDTCUT(15),ITCUT(15)
      COMMON /PARM1/ TSTD(12),PW(12),WN(6),CSTD(6),NCUT,NC,NAME(20),
     *               AN,AM,CF,ICONST(6),NEL
      COMMON /PARM3/ A1(11),A2(11),A3(11)
      SSD=0.
      ITOTAL=0
      IM=NCUT-1
      JM=NCUT-2
C
C
C        COMPUTATION OF THE COEFFICIENTS A1(I), A2(I) AND A3(I)
C
      CTX1=ALOG10(-ALOG(TSTD(1)))
      DO 50 I=1,IM
      PDIF=PW(I)-PW(I+1)
      CTX2=ALOG10(-ALOG(TSTD(I+1)))
      A1(I)=(PW(I)*CTX2-PW(I+1)*CTX1)/PDIF
      A2(I)=(CTX1-CTX2)/PDIF
      A3(I)=0.
      CTX1=CTX2
C
      SDTCUT(I)=0.
   50 ITCUT(I)=0
C
C        THE FIRST AND LAST VALUES OF TSTD AND PW ARE CHANGED
C        FOR THE TABLE OUTPUT. TRUE VALUES ARE TEMPORARY STORED
C        IN THE RESERVE.
C
      TRES1=TSTD(1)
      TRES2=TSTD(NCUT)
      PWRES1=PW(1)
      PWRES2=PW(NCUT)
      TSTD(1)=1.0
      TSTD(NCUT)=0.0
```

```
      PW(1)=-1.E70
      PW(NCUT)=1.E70
C
C         PRINT OUT OF THE RESULTS
C
      WRITE(6,2) (NAME(I),I=1,20)
    2 FORMAT(1H1,T25,20A4,//,T15,'PIECEWISE-ANALYTICAL STANDARD',
     * 'TRANSMISSION FUNCTION',//,T20,'TAU(X) =',
     * ' EXP(-10**( A1 + A2*X ) )',///,T15,'DATA:',T23,'FROM ( TAU  ,',
     * 'X-VALUE) TO (  TAU  ,X-VALUE)  WITH  (  A1   ,   A2  )')
      WRITE(6,3) (TSTD(I),PW(I),TSTD(I+1),PW(I+1),A1(I),A2(I),I=1,IM)
    3 FORMAT(1H0,T28,'(',F6.3,' ,',F7.3,')    (',F6.3,' ,',F7.3,')',T74,
     * '('.F7.4,',',F7.4,')')
      WRITE(6,4) (I,WN(I),CSTD(I),I=1,NC)
    4 FORMAT(1H0,//,T15,'ABSORPTION BANDS:',T40,
     * '#  WAVENUMBER   C-VALUE'/(1H0,T39,I2,5X,F7.1,F11.5))
C
C         CHECK IF ANY DATA IS AVAILABLE FOR S.D. COMPUTATION
C         DATA FORMAT IS THE SAME AS THAT FOR MAIN PROGRAMME
C         ONE CONTROL CARD IS READ-IN FIRST FOR BRANCHING
C             IFQ >  0   DATA SET FOLLOWS, READ-IN DATA
C             IFQ =  0   END OF DATA, GO TO THE FINAL PRINTING
C
      READ(5,11,END=42) FQ,IFQ
   11 FORMAT(5X,F10.3,T41,I4)
      IF(IFQ.GT.0) GO TO 18
   42 WRITE(6,41)
   41 FORMAT(///,1H0,T5,'$$$  NO DATA FOR STANDARD DEVIATION COMPUTATION
     *  $$$')
      GO TO 40
    8 READ(5,11,END=30) FQ,IFQ
      IF(IFQ.LE.0) GO TO 30
   18 CONTINUE
      ST=0.
      DO 51 I=1,IM
      SDCUT(I)=0.
   51 ICUT(I)=0
      CLOG=100.
      DO 52 I=1,NC
   52 IF(ABS(FQ-WN(I)).LT.1.) CLOG=CSTD(I)
      IF(CLOG.LT.99.) GO TO 13
C
C         THE READ-IN WAVENUMBER DOES NOT MATCH THE MAJOR BAND
C         WAVENUMBER (WN(I)).THE DATA IN THIS BAND ARE IGNORED.
C
      WRITE(6,12) FQ
   12 FORMAT(1H0,T10,'** ERROR IN WAVENUMBER **',
     * '      (READ-IN WAVENUMBER =',F10.5,' )')
      DO 61 IDUM=1,IFQ
      READ(5,60) DUMMY
   60 FORMAT(F1.0)
   61 CONTINUE
      GO TO 8
C
C         VALID DATA INPUTS,READ-IN OF THE DATA AND STANDARD
C         DEVIATION COMPUTATION ARE PERFORMED SIMULTANEOUSLY.
```

```
C
   13 CONTINUE
      WRITE(6,17) FQ
   17 FORMAT(1H1,T15,'( WAVE NUMBER =',F8.1,' )',///,6X,'WAVEN.',3X,
     * 'PRESS.',4X,'TEMP.',7X,'U',8X,'TRANSM.  -  T(COMP)  =   DIFF',6X,
     * 'DIFF**2',4X,'X-VALUE',/)
      DO 9 M=1,NDATA
      READ(5,10) KGAS,FQ,PRES,TEMP,UG,TX
   10 FORMAT(I2,F10.3,E11.4,F9.3,24X,E11.4,F7.4)
      UG=UG/CF
      X=CLOG+AN*ALOG10(PRES/1013.)+AM*ALOG10(273.15/TEMP)+ALOG10(UG)
      DO 14 J=1,JM
      IF(X.LE.PW(J+1)) GO TO 15
   14 CONTINUE
      J=IM
C
   15 TC=EXP(-10**(A1(J)+A2(J)*X))
C
      D=TX-TC
      SD=D*D
      ST=ST+SD
      SDCUT(J)=SDCUT(J)+SD
      ICUT(J)=ICUT(J)+1
      WRITE(6,16) FQ,PRES,TEMP,UG,TX,TC,D,SD,X
   16 FORMAT(1H ,3X,F8.1,F10.2,F9.2,E13.4,F9.4,F12.4,F13.6,E12.3,F9.3)
    9 CONTINUE
C
C         END OF DATA READ-IN FOR THIS BAND.
C         TOTAL STANDARD DEVIATIONS ARE COMPUTED AND PRINTED.
C
   20 SSD=SSD+ST
      ITOTAL=ITOTAL+IFQ
      ST=SQRT(ST/FLOAT(IFQ))
      DO 21 I=1,IM
      SDTCUT(I)=SDTCUT(I)+SDCUT(I)
      ITCUT(I)=ITCUT(I)+ICUT(I)
   21 SDCUT(I)=SQRT(SDCUT(I)/FLOAT(ICUT(I)))
      WRITE(6,22) (I,TSTD(I),TSTD(I+1),ICUT(I),SDCUT(I),I=1,IM)
   22 FORMAT(1H0,///,T10,'CUTWISE STANDARD DEVIATION',//,T15,'#',T20,
     * '( FROM,  TO )  ',T40,'# OF DATA',T53,'CUTWISE SD',//,(T14,I2,
     * T20,'(',F5.2,',',F5.2,')',T43,I4,T52,F10.6,/))
      WRITE(6,23) IFQ,ST
   23 FORMAT(1H0,T10,'TOTAL # OF DATA FOR THIS BAND  =',I5,9X,
     * ' STANDARD DEVIATION =',F12.6,//)
      GO TO 8
C
C         END OF THE STANDARD DEVIATION COMPUTATION FOR ALL DATA.
C         GRAND TOTAL STANDARD DEVIATION IS COMPUTED AND PRINTED OUT
C         TOGETHER WITH VITAL INFORMATIONS.
C
   30 SSD=SQRT(SSD/FLOAT(ITOTAL))
      DO 31 I=1,IM
   31 SDTCUT(I)=SQRT(SDTCUT(I)/FLOAT(ITCUT(I)))
      WRITE(6,32) (I,TSTD(I),PW(I),TSTD(I+1),PW(I+1),A1(I),A2(I),
     * ITCUT(I),SDTCUT(I),I=1,IM)
   32 FORMAT(1H1,T20,'***  PIECEWISE-ANALYTICAL STANDARD TRANSMISSION',
```

```
     * ' FUNCTION  ***',///,T10,'TOTAL CUTWISE STANDARD DEVIATION',//,
     * T15,'CURVE #',3X,'FROM (   TAU   ,X-VALUE) TO (   TAU   ,X-VALUE)',
     * '   WITH  (    A1    ,    A2   )',3X,'# OF DATA',4X,'CUTWISE SD',//,
     * (T18,I2,T30,'(',F6.3,' ,',F7.3,')    (',F6.3,' ,',F7.3,')',T76,
     * '(',F7.4,',',F7.4,')',7X,I3,5X,F10.6,/))
      WRITE(6,33) ITOTAL,SSD
   33 FORMAT(1H0,T10,'GLOBAL RESULTS',//,T15,'TOTAL NUMBER OF DATA',
     * ' USED',I5,//,T15,'GLOBAL STANDARD DEVIATION',F12.6)
C
   40 CONTINUE
C
C        END OF ALL COMPUTATION.
C        RESERVED TRUE VALUES OF THE FIRST AND LAST TSTD
C        AND PW ARE RETURNED.
C
      TSTD(1)=TRES1
      TSTD(NCUT)=TRES2
      PW(1)=PWRES1
      PW(NCUT)=PWRES2
C
      CALL SDTAU
C
      RETURN
      END
       SUBROUTINE INTPL2
C
C         COMPUTATION OF THE PIECEWISE-ANALYTICAL STANDARD TRANSMISSION
C         FUNCTION
C
C         VERSION 2 - 1
C         TAU=EXP(-10**(A1+A2*X+A3*X**2))
C
      COMMON /PARM1/ TSTD(12),PW(12),WN(6),CSTD(6),NCUT,NC,NAME(20),
     *               AN,AM,CF,ICONST(6),NEL
      COMMON /PARM3/ A1(11),A2(11),A3(11)
      DIMENSION T(10),TD(6,74),PD(6,74),JI(6,10),
     * SDK(7),SDE(7,9),SUME(2,9),DE(6,9)
      NWC=0
      K=0
      READ(5,2,END=80) FREQ,MAXDAT
      IF(MAXDAT.GT.0) GO TO 3
   80 CONTINUE
      WRITE(6,99)
   99 FORMAT(///,1H0,T5,'$$$  NO DATA FOR STANDARD DEVIATION COMPUTATION
     *  $$$')
      GO TO 77
    1 CONTINUE
      READ(5,2,END=21) FREQ,MAXDAT
    2 FORMAT(5X,F10.5,T41,I4)
      IF(MAXDAT.EQ.0) GO TO 21
    3 CLOG=1.E 10
      DO 5 L=1,NC
      IF(ABS(FREQ-WN(L)).LE. .01)  CLOG=CSTD(L)
    5 CONTINUE
      IF(ABS(CLOG-1.E10).GE..01) GO TO 9
      WRITE(6,100) FREQ
```

```
100 FORMAT('1',///////,' ERROR IN INPUT DATA ; WAVE NUMBER ',F10.3,
   * ' NOT USED IN COMPUTATION OF CONSTANTS.')
    DO 6 J=1,MAXDAT
    READ(5,101) KGAS,FREQ,PRES,TEMP,PPM,RANGE,UGAS,TX
  6 CONTINUE
    GO TO 1
  9 CONTINUE
    K=K+1
    NWC=NWC+1
    JI(K,1)=0
    JI(K,NCUT)=MAXDAT
    J=2
    DO 20 I=1,MAXDAT
    READ(5,101) KGAS,FREQ,PRES,TEMP,PPM,RANGE,UGAS,TX
101 FORMAT(T2,F10.3,E11.4,F9.3,E11.4,E13.6,E11.4,F7.4)
    TD(K,I)=TX
    PD(K,I)= AN *ALOG10(PRES/1013.)+ AM *ALOG10(273.15/TEMP)+ALOG10
   * (UGAS/CF)+CLOG
    IF(TD(K,I).GE.TSTD(J)) GO TO 20
    IF(J.EQ.NCUT) GO TO 20
    JI(K,J)=I-1
    J=J+1
 20 CONTINUE
    GO TO 1
 21 CONTINUE
    IF(NWC.LE.0) RETURN
    DO 30 J=1,NCUT
    T(J)=ALOG10(-ALOG(TSTD(J)))
 30 CONTINUE
    SUMT=0.
    DT=0.0
    NCC=NCUT-1
    DO 45 I=1,NCC
    SA=0.
    TA=0.
    UA=0.
    DO 41 K=1,NWC
    SUME(K,I)=0.0
    M=JI(K,I)+1
    N=JI(K,I+1)
    DO 40 J=M,N
       TC=ALOG10(-ALOG(TD(K,J)))
    SA=SA+ TC  *PW(I)*PW(I+1)*(PD(K,J)-PW(I))*(PD(K,J)-PW(I+1))
    TA=TA+((PD(K,J)-PW(I))*(PD(K,J)-PW(I+1))*((PD(K,J)**2)*(PW(I+1)
   * *T(I)-PW(I)*T(I+1))+PD(K,J)*((PW(I)**2)*T(I+1)-(PW(I+1)**2)*T(I)
   * )))/(PW(I)-PW(I+1))
    UA=UA+((PD(K,J)-PW(I))*(PD(K,J)-PW(I+1)))**2
 40 CONTINUE
 41 CONTINUE
    A1(I)=(SA-TA)/UA
    A3(I)=(T(I)-T(I+1))/(PW(I+1)*(PW(I)-PW(I+1)))-(T(I)-A1(I))/(PW(I)*
   * PW(I+1))
    A2(I)=(T(I)-T(I+1))/(PW(I)-PW(I+1))-A3(I)*(PW(I)+PW(I+1))
      [illegible]
      [illegible] K=1,NWC
```

```
      M=JI(K,I)+1
      N=JI(K,I+1)
      DE(K,I)=FLOAT(1+N-M)
      DO 43 J=M,N
      SUME(K,I)=SUME(K,I)+(TD(K,J)-EXP(-10.**(A3(I)*PD(K,J)*PD(K,J)+
     *  A2(I)*PD(K,J)+A1(I))))**2
   43 CONTINUE
      SDE(K,I)=SQRT(SUME(K,I)/DE(K,I))
      SUMI=SUMI+SUME(K,I)*FLOAT(ICONST(K))
      DI=DI+DE(K,I)*FLOAT(ICONST(K))
   44 CONTINUE
      SUMT=SUMT+SUMI
      DT=DT+DI
      SDE(NWC+1,I)=SQRT(SUMI/DI)
   45 CONTINUE
      DO 51 K=1,NWC
      SUMK=0.0
      DK=0.0
      DO 50 I=1,NCC
      SUMK=SUMK+SUME(K,I)
      DK=DK+DE(K,I)
   50 CONTINUE
      SDK(K)=SQRT(SUMK/DK)
   51 CONTINUE
      SDK(NWC+1)=SQRT(SUMT/DT)
      DUM1=PW(1)
      DUM2=PW(NCUT)
      DUM3=TSTD(1)
      DUM4=TSTD(NCUT)
      PW(1)=-1000000.
      PW(NCUT)=1000000.
      TSTD(1)=1.0
      TSTD(NCUT)=0.0
      DO 60 K=1,NWC
      WRITE(6,102)(NAME(J),J=1,20),WN(K),NCC,(I,TSTD(I),TSTD(I+1),PW(I),
     * PW(I+1),A1(I),A2(I),A3(I),SDE(K,I),I=1,NCC)
  102 FORMAT  ('1',/,35X,'RENDITION OF EMPIRICAL TRANSMITTANCE FUNCTION
     *FOR : '//,20A4,//,40X,'WAVE  NUMBER :',F15.4,////,20X,'THE TRANSMI
     *SSION CURVE IS DIVIDED INTO',I3,'  SEPARATE CURVES.',/20X,'EACH CU
     *RVE IS EXPRESSED BY A FUNCTION OF THE FORM  " TAU = EXP(-10**(A3*P
     #*P+A2*P+A1)) ".'/,20X,                        'THE  FUNCTION  COEFFI
     *CIENTS AND RESULTING STANDARD DEVIATION  FOR EACH CURVE ARE AS FOL
     *LOWS:',////22X,'TAU',20X,'P',24X,'A1',13X,'A2',13X,'A3',11X,'STAND
     *ARD DEVIATION',///, (5X,'CURVE #',I3,3X,'(',F4.2,'-',F4.2,')',5X,
     *'(',F9.5,'-',F9.5,')',5X,3F15.6, 6X,F15.6 /))
      WRITE(6,401) SDK(K)
  401 FORMAT(1X,//,87X,'TOTAL STANDARD DEVIATION',F15.6)
   60 CONTINUE
      K=K+1
      WRITE(6,104)  (NAME(J),J=1,20),NCC,  (I,TSTD(I),TSTD(I+1),PW(I),
     * PW(I+1),A1(I),A2(I),A3(I),SDE(K,I),I=1,NCC)
  104 FORMAT  ('1',/,35X,'RENDITION OF EMPIRICAL TRANSMITTANCE FUNCTION
     *FOR : '//,20A4,//,40X,'TOTAL PROFILE AVERAGED OVER ALL WAVE NUMBER
     *S'                                          ,////,20X,'THE TRANSMI
     *SSION CURVE IS DIVIDED INTO',I3,'  SEPARATE CURVES.',/20X,'EACH CU
     *RVE IS EXPRESSED BY A FUNCTION OF THE FORM  " TAU = EXP(-10**(A3*P
```

```
     #*P+A2*P+A1)) ".'/,20X,                        'THE  FUNCTION  COEFFI
     *CIENTS AND RESULTING STANDARD DEVIATION  FOR EACH CURVE ARE AS FOL
     *LOWS:',////22X,'TAU',20X,'P',24X,'A1',13X,'A2',13X,'A3',11X,'STAND
     *ARD DEVIATION',///, (5X,'CURVE #',I3,3X,'(',F4.2,'-',F4.2,')',5X,
     *'(',F9.5,'-',F9.5,')',5X,3F15.6, 6X,F15.6 /))
      WRITE(6,402) SDK(K)
  402 FORMAT(1X,//,81X,'GRAND TOTAL STANDARD DEVIATION',F15.6)
C     WRITE(7,201)(A1(I),A2(I),A3(I),I=1,NCC)
C 201 FORMAT(3F10.6)
C
      IDT=IFIX(DT)
      WRITE(6,225)IDT,NEL,SDK(K)
  225 FORMAT(//,1H0,T15,'GRAND TOTAL # OF DATA =',I5,//,T15,'# OF',
     * ' ELIMINATED DATA =',I5,//,T15,'GLOBAL STANDARD DEVIATION IN',
     * ' TAU =',F12.6,//)
      DO 76 M=1,NWC
      IF(ICONST(M).EQ.1) GO TO 76
      WRITE(6,226) M
  226 FORMAT(1H ,T15,'NOTE:  THE BAND',I3,' IS NOT INCLUDED IN THE',
     * ' FINAL STANDARD DEVIATION')
   76 CONTINUE
      PW(1)=DUM1
      PW(NCUT)=DUM2
      TSTD(1)=DUM3
      TSTD(NCUT)=DUM4
C
      CALL SDTAU
C
   77 CONTINUE
      RETURN
      END
      SUBROUTINE SDTAU
C
C        COMPUTATIONS OF STANDARD DEVIATIONS IN TAU USING THE ORIGINAL
C        DATA USED IN MAIN
C
      DIMENSION NDATA(6),TSD(6),WWW(12,10),STANDV(12),P(10),T(10)
      COMMON /PARM1/ TSTD(12),PW(12),WN(6),CSTD(6),NCUT,NC,NAME(20),
     *               AN,AM,CF,ICONST(6),NEL
      COMMON /PARM2/ PRES(6,12,10),TEMP(6,12,10),UGAS(6,12,10),
     *               TAU(6,12),NTC(6),NLV(6)
      COMMON /PARM3/ A1(11),A2(11),A3(11)
C
      NGDATA=0
      GTSD=0.
      ICST=NC
      DO 70 M=1,NC
      JM=NTC(M)
      KM=NLV(M)
      NDATA(M)=JM*KM
      NGDATA=NGDATA+NDATA(M)*ICONST(M)
      TSD(M)=0.
      WRITE(6,214)
  214 FORMAT('1',/////,45X,'RECOMPUTATION OF TAU',/////)
      IF(M.GT.1) GO TO 77
       WRITE(6,215)
```

```
  215 FORMAT(20X,'A TAU VALUE , T , IS RECOMPUTED FOR THE ORIGIONAL DATA
     * USING THE PIECEWISE-ANALITICAL TRANSMISSION FUNCTION.'//20X,
     * 'STANDARD DEVIATIONS BETWEEN THE ACTUAL TAU AND THE RECOMPUTED',
     * ' TAU VALUES ARE COMPUTED.'////)
   77 CONTINUE
      WRITE(6,202) M,WN(M),NTC(M),NLV(M)
  202 FORMAT(1H0,T15,'***  CASE',I3,'  (WAVE NUMBER =',F10.3,')  ***',
     * ///,T20,'TOTAL # OF CUTS   =',I3,//,T20,'TOTAL # OF LEVELS =',I3,
     * ///)
      WRITE(6,216) AN,AM,M,CSTD(M)
  216 FORMAT(10X,'N  =',F10.5,//10X,'M  =',F10.5,//,10X,'C',I1,' =',
     * F10.5,////)
      WRITE(6,217)
  217 FORMAT(//,1H0,T7,'RECOMPUTED TAU AND STANDARD DEVIATIONS',
     * ' IN TAU ',/,1H0,T2,'CUT',T11,'TAU',T20,'X*',T30,'X1',T39,
     * 'X2',T48,'X3',T57,'X4',T66,'X5',T75,'X6',T84,'X7',T93,'X8',
     * T102,'X9',T111,'X10',T121,'CUTWISE-SD',/)
C
C        COMPUTATION OF THE CUTWISE STANDARD DEVIATIONS IN X
C
      DO 71 J=1,JM
      WW=0.
      DO 72 K=1,KM
      P(K)=CSTD(M)+AN*PRES(M,J,K)+AM*TEMP(M,J,K)+UGAS(M,J,K)
      IM=JM-1
      DO 75 I=1,IM
      IF(PW(I+1).GT.P(K)) GO TO 76
   75 CONTINUE
      I=IM
   76 CONTINUE
      T(K)=EXP(-10**(A3(I)*P(K)*P(K)+A2(I)*P(K)+A1(I)))
      WWW(J,K)=(TAU(M,J)-T(K))**2
      WW=WW+WWW(J,K)
   72 CONTINUE
      WW=SQRT(WW/FLOAT(KM))
      WRITE(6,218) J,PW(J),TAU(M,J),(T(K),K=1,KM)
  218 FORMAT(1H ,I5,F9.4,F9.4,1X,10F9.4)
      WRITE(6,219) WW
  219 FORMAT(1H+,T121,F10.5)
   71 CONTINUE
C
C        COMPUTATION OF THE LEVELWISE STANDARD DEVIATIONS IN X
C
      DO 73 K=1,KM
      WW=0.
      DO 74 J=1,JM
      WW=WW+WWW(J,K)
   74 CONTINUE
      TSD(M)=TSD(M)+WW
      STANDV(K)=SQRT(WW/FLOAT(JM))
   73 CONTINUE
      WRITE(6,220) (STANDV(K),K=1,KM)
  220 FORMAT(1H0,T4,'LEVELWISE-SD :',T26,10F9.5)
      GTSD=GTSD+TSD(M)*FLOAT(ICONST(M))
      ICST=ICST-ICONST(M)
      TSD(M)=SQRT(TSD(M)/FLOAT(NDATA(M)))
```

```
      WRITE(6,221) TSD(M)
  221 FORMAT(1H0,//,T15,'TOTAL STANDARD DEVIATION FOR THIS CASE  :',
     * F15.6)
   70 CONTINUE
      GTSD=SQRT(GTSD/FLOAT(NGDATA))
      WRITE(6,223) AN,AM
  223 FORMAT('1',T15,'***  SUMMARY OF THE TRANSMITTANCE RECOMPUTATION  *
     * **',///,T20,'PRESSURE EXPONENT    N =',F10.5,//,T20,
     * 'TEMPERATURE EXPONENT M =',F10.5,///,T5,'CASE #',3X,
     * 'WAVE NUMBER',5X,'C-VALUE',5X,'TOTAL # OF DATA',3X,
     * 'CASEWISE S.D. IN TAU')
      WRITE(6,224) (M,WN(M),CSTD(M),NDATA(M),TSD(M),M=1,NC)
  224 FORMAT(1H0,T6,I3,6X,F9.2,5X,F8.3,10X,I3,12X,F12.6)
      WRITE(6,225) NGDATA,NEL,GTSD
  225 FORMAT(//,1H0,T15,'GRAND TOTAL # OF DATA =',I5,//,T15,'# OF',
     * ' ELIMINATED DATA =',I5,//,T15,'GLOBAL STANDARD DEVIATION IN',
     * ' TAU =',F12.6,//)
      IF(ICST.LE.0) RETURN
      DO 78 M=1,NC
      IF(ICONST(M).EQ.1) GO TO 78
      WRITE(6,226) M
  226 FORMAT(1H ,T15,'NOTE:  THE BAND',I3,' IS NOT INCLUDED IN THE',
     * ' FINAL STANDARD DEVIATION')
   78 CONTINUE
      RETURN
      END
```

```
C
C******  COMPUTER CODE  SIMMIN  ******
C        VERSION ( 6 - 3 ) TRACE GASSES
C
C        COMPUTATION OF ABSORBER PARAMETERS AND ANALYTICAL STANDARD
C        TRANSMISSION FUNCTION
C
C        THIS CODE USES THE SUBROUTINE FMCG IN SSP LIBRARY
C
C     THIS CODE CONSISTS OF
C        MAIN:  DATA READ-IN AND CONTROL OF COMPUTATION
C        FUNCT:  COMPUTATION OF THE COST AND ITS DERIVATIVES
C        TITLE:  PRINTING OF HEADINGS AND INITIAL CONDITIONS
C        PRINT1:  PRINTOUT OF RESULTS AND COMPUTATION/PRINTING OF S.D.S
C        NMBC:  COMPUTATION OF NON-MAJOR BANDS' C-VALUES
C
C     DATA SET-UP
C        1. INITIAL GUESSES  X(I)  (9 CARDS WITH T12,F10.7)
C           X(1)=A1, X(2)=A2, X(3)=A3, X(4)=N, X(5)=M, X(5+I)=LOG(C(I))
C           (NEED DUMMY INPUTS FOR PROBLEMS WITH DIMENSION < 9)
C        2. SIGNAL VARIABLES  S(I)  (9 CARDS WITH T12,F10.7)
C            S(I) = 0  ...  X(I) IS KEPT CONSTANT
C            S(I) = 1  ...  X(I) IS VARIED
C            (NEED DUMMY INPUTS FOR PROBLEMS WITH DIMENSION < 9)
C        3. COMMENT CARD  (20A4)  FOR TITLE AND ABSORBER TYPE ETC.
C        4. DATA SETS (MAX. 4 SETS)  -  ONE FOR EACH ABSORPTION BAND
C            EACH SET CONSISTS OF
C               1ST(CONTROL) CARD: WAVENUMBER, # OF DATA AND COMMENTS
C                                  (SEE FORMAT 101)
C               DATA CARDS: P, T, U, TAU ETC.
C                                  (SEE FORMAT 102)
C            (TOTAL # OF DATA SHOULD NOT EXCEED 900)
C        5. BLANK CARD  -  FOR THE TERMINATION OF DATA INPUT FOR MAIN
C        6. DATA SETS FOR NMBC  -  ONE FOR EACH ABSORPTION BAND
C            EACH SET CONSISTS OF
C               DATA CARDS:  SAME AS MAIN
C               FINAL CARD:  BLANK
C            TERMINATION   A CONTROL CARD WITH -1 IN FIRST TWO COLUMNS
C                          THIS COMES AFTER THE FINAL BLANK CARD
C            (IF NO DATA BUT A BLANK CARD IS SUPPLIED, NMBC IS SKIPPED)
C
      DIMENSION X(9),G(9),Y(9),H(72),WN(4)
      COMMON /PARM1/ NC,ND(5),RW(4)
      COMMON /PARM2/ IC,PLOG(900),TLOG(900),ULOG(900),TAU(900),S(9)
      COMMON /PARM3/ P(900),T(900),U(900),L(20)
      COMMON /PARM4/ P0,T0,NDIM,ID(5,9)
      EXTERNAL FUNCT
C
C        CONSTANTS
      P0=1.013E+03
      T0=273.15
      CF=2.69E+19
      N=9
      V=0.
      IC=0
      MAXNC=4
```

```
C
C         DATA INPUT
C
      READ(5,100) (X(I),I=1,N)
      READ(5,100) (S(I),I=1,N)
  100 FORMAT (T12,F10.7)
C
C         COMMENT CARD (THIS INCLUDES THE ABSORBER TYPE)
      READ(5,500) (L(I),I=1,20)
  500 FORMAT(20A4)
C
C         NC = # OF MAJOR ABSORPTION BANDS
C         ND(1)=0, ND(2)=N1, ND(3)=N1+N2, ND(4)=N1+N2+N3, ...
C         WHERE N1, N2, N3, ... ARE #S OF DATA IN BANDS 1, 2, 3, ...
C         ND(NC+1)= TOTAL # OF DATA
C
      NC=0
      ND(1)=0
      DO 10 M=1,MAXNC
      READ(5,101) WN(M),IX,(ID(NC+1,I),I=1,9)
  101 FORMAT(5X,F10.3,T41,I4,9A4)
      IF(IX.LE.0) GO TO 11
      NC=NC+1
      IM=ND(NC)+1
      IN=ND(NC)+IX
      ND(NC+1)=IN
      DO 12 I=IM,IN
      READ(5,102) P(I),T(I),U(I),TAU(I)
  102 FORMAT (12X,E11.4,F9.3,24X,E11.4,F7.4)
      U(I)=U(I)/CF
C
C         DATA ARE CONVERTED TO THE LOG OF THE NORMALIZED VALUES
C
      PLOG(I)=ALOG10(P(I)/P0)
      TLOG(I)=ALOG10(T0/T(I))
      ULOG(I)=ALOG10(U(I))
C
   12 CONTINUE
   10 CONTINUE
   11 CONTINUE
C
C         END OF DATA INPUT
C
      IF(NC.GT.0) GO TO 13
      WRITE(6,110)
  110 FORMAT (1H0,'ERROR IN DATA INPUT')
      GO TO 1000
   13 CONTINUE
C
      NDIM=0
      N=5+NC
      DO 14 I=1,N
      IF(S(I).NE.0.) NDIM=NDIM+1
   14 CONTINUE
      DO 15 I=1,NC
      RW(I)=FLOAT(ND(NC+1))/(FLOAT(ND(I+1)-ND(I))*FLOAT(NC))
```

```
      RW(I)=1.0
   15 CONTINUE
C
      DO 16 I=1,N
      Y(I)=X(I)
   16 CONTINUE
C
      EST=1.E-6
      EPS=1.E-6
      LIMIT=1
      IER=0
C
C************* FMCG SEARCH *********************** START ******************
C
      CALL FMCG(FUNCT,N,X,V,G,EST,EPS,LIMIT,IER,H)
C
C************* FMCG SEARCH ***********************  END  ******************
C
      CALL TITLE(N,Y,LIMIT,EPS)
C
      CALL PRINT1(N,X,V,G,IER)
C
      CALL NMBC(X,L,NC,WN,CF,PO,TO)
C
 1000 CONTINUE
      STOP
      END
      SUBROUTINE FUNCT(N,X,V,G)
C
C        COMPUTATION OF THE FUNCTION VALUE AND DERIVATIVES
C
C          (DOUBLE EXPONENTIAL FUNCTION)
C
      DIMENSION X(9),G(9),F(9)
      COMMON /PARM1/ NC,ND(5),RW(4)
      COMMON /PARM2/ IC,PLOG(900),TLOG(900),ULOG(900),TAU(900),S(9)
C
      IC=IC+1
      V=0.
      DO 20 K=1,N
      G(K)=0.
   20 CONTINUE
C
      DO 21 J=1,NC
      JJ=J+5
      SQER=0.
      DO 22 L=1,5
      F(L)=0.
   22 CONTINUE
      F(JJ)=0.
      IM=ND(J)+1
      IN=ND(J+1)
      DO 23 I=IM,IN
      W1=X(JJ)+X(4)*PLOG(I)+X(5)*TLOG(I)+ULOG(I)
      R=X(1)+X(2)*W1+X(3)*W1*W1
      R=10.**R
```

```
      IF(R.LE.70.) GO TO 24
      TC=0.
      GO TO 25
   24 CONTINUE
      TC=EXP(-R)
   25 CONTINUE
      E=TAU(I)-TC
      R=R*E*TC
      SQER=SQER+E**2
      F(1)=F(1)+R
      F(2)=F(2)+R*W1
      F(3)=F(3)+R*W1*W1
      R=R*(X(2)+2.*X(3)*W1)
      F(4)=F(4)+R*PLOG(I)
      F(5)=F(5)+R*TLOG(I)
      F(JJ)=F(JJ)+R
   23 CONTINUE
      V=V+SQER*RW(J)
      DO 26 K=1,5
      G(K)=G(K)+F(K)*RW(J)
   26 CONTINUE
      G(JJ)=F(JJ)*RW(J)
   21 CONTINUE
C
      DO 27 I=1,N
      G(I)=4.60517*G(I)*S(I)
   27 CONTINUE
C
      RETURN
      END
      SUBROUTINE TITLE(N,X,LIMIT,EPS)
C
C        PRINTING OF THE TITLE AND INITIAL VALUES
C
      DIMENSION X(9),L(4)
      COMMON /PARM1/ NC,ND(5),RW(4)
      COMMON /PARM4/ PO,TO,NDIM,ID(5,9)
C
      DO 40 I=1,NC
      L(I)=ND(I+1)-ND(I)
   40 CONTINUE
C
      CALL DATE (MONTH,IDAY,IYEAR)
      WRITE(6,111)MONTH,IDAY,IYEAR
  111 FORMAT(1H1,T60,I4,' / ',I2,' / ',I2,/)
      WRITE(6,400) NC,NDIM
  400 FORMAT (1H ,T14,'*** SIMULTANEOUS PARAMETER EVALUATION ***',///,
     * ' PARAMETERS  :  ( N , M , A1 , A2 , A3 , C(I), I=1,',I2,' )',
     * 8X,'( DIMENSION =',I3,' )',///,' DATA  :')
      WRITE(6,401) ((ID(K,J),J=1,9),L(K),K=1,NC)
  401 FORMAT(1H+,T11,'(',9A4,' )',5X,'# OF POINTS =',I5,//)
C
      WRITE(6,402) ND(NC+1),PO,TO,LIMIT,EPS
  402 FORMAT (1H+,T51,'TOTAL # OF DATA =',I5,///,
     * ' FUNCTION  :  TAU ( W )  = EXP ( -10 ** ( A1 + A2 * W +',
     * ' A3 * W**2 + A4 * W**3 ) )',//,T15,'WHERE, ',
```

```
     * ' W = LOG(C) + LOG( U * (P/PO)**N * (TO/T)**M )  ',
     * ',  A4 = 0.',///,
     * ' CONSTANTS  :  PO =',F8.2,7X,'TO =',F8.2,7X,
     * 'LIMIT =',I5,7X,'EPS =',1PE10.1,/)
C
      IF(NC.EQ.1) WRITE(6,403) X(1),X(6),X(2),X(3),X(4),X(5)
      IF(NC.EQ.2) WRITE(6,404) X(1),X(6),RW(1),X(2),X(7),RW(2),X(3),
     *                         X(4),X(5)
      IF(NC.EQ.3) WRITE(6,405) X(1),X(6),RW(1),X(2),X(7),RW(2),X(3),
     *                         X(8),RW(3),X(4),X(5)
      IF(NC.EQ.4) WRITE(6,406) X(1),X(6),RW(1),X(2),X(7),RW(2),X(3),
     *                         X(8),RW(3),X(4),X(9),RW(4),X(5)
  403 FORMAT (1H0,'INITIAL VALUES  :',T22,'A1 =',F12.7,9X,'LOG(C1) =',
     * F12.7,//,T22,'A2 =',F12.7,//,T22,'A3 =',F12.7,//,T22,'N  =',
     * F12.7,//,T22,'M  =',F12.7,/)
  404 FORMAT (1H0,'INITIAL VALUES  :',T22,'A1 =',F12.7,9X,'LOG(C1) =',
     * F12.7,4X,'( WEIGHT =',F8.4,' )',//,T22,'A2 =',F12.7,9X,
     * 'LOG(C2) =',F12.7,4X,'( WEIGHT =',F8.4,' )',//,T22,'A3 =',
     * F12.7,//,T22,'N  =',F12.7,//,T22,'M  =',F12.7,/)
  405 FORMAT (1H0,'INITIAL VALUES  :',T22,'A1 =',F12.7,9X,'LOG(C1) =',
     * F12.7,4X,'( WEIGHT =',F8.4,' )',//,T22,'A2 =',F12.7,9X,
     * 'LOG(C2) =',F12.7,4X,'( WEIGHT =',F8.4,' )',//,T22,'A3 =',
     * F12.7,9X,'LOG(C3) =',F12.7,4X,'( WEIGHT =',F8.4,' )',//,T22,
     * 'N  =',F12.7,//,T22,'M  =',F12.7,/)
  406 FORMAT (1H0,'INITIAL VALUES  :',T22,'A1 =',F12.7,9X,'LOG(C1) =',
     * F12.7,4X,'( WEIGHT =',F8.4,' )',//,T22,'A2 =',F12.7,9X,
     * 'LOG(C2) =',F12.7,4X,'( WEIGHT =',F8.4,' )',//,T22,'A3 =',
     * F12.7,9X,'LOG(C3) =',F12.7,4X,'( WEIGHT =',F8.4,' )',//,T22,
     * 'N  =',F12.7,9X,'LOG(C4) =',F12.7,4X,'( WEIGHT =',F8.4,' )',//,
     * T22,'M  =',F12.7,/)
C
      RETURN
      END
      SUBROUTINE PRINT1(N,X,V,G,IER)
C
C        PRINTING OF THE RESULTS AND COMPUTATION/PRINTING OF ERRORS
C        AND STANDARD DEVIATIONS
C
      DIMENSION X(9),G(9),E(900),PD(900),TC(900),W(900)
      COMMON /PARM1/ NC,ND(5),RW(4)
      COMMON /PARM2/ IC,PLOG(900),TLOG(900),ULOG(900),TAU(900),S(9)
      COMMON /PARM3/ P(900),T(900),U(900),L(20)
      COMMON /PARM4/ PO,TO,NDIM,ID(5,9)
      EQUIVALENCE (E,PLOG),(PD,TLOG),(TC,ULOG)
C
C        EQUIVALENCE IS FOR SPACE CONSERVATION
C
      IT=0
      TTD=0.
      TV=0.
      V=SQRT(V/FLOAT(ND(NC+1)))
C
      WRITE(6,510) IER,IC,(X(I),G(I),I=1,5)
  510 FORMAT (1H0,'** RESULTS OF COMPUTATION  :   IER =',I3,4X,
     * 'SUBROUTINE FUNCT WAS CALLED',I6,' TIMES  **',
     * ///,' FINAL VALUES AND GRADIENTS  :',
```

```
     * T35,'A1      =',F12.7,T65,'D/D(A1)      =',E15.6,//,
     * T35,'A2      =',F12.7,T65,'D/D(A2)      =',E15.6,//,
     * T35,'A3      =',F12.7,T65,'D/D(A3)      =',E15.6,//,
     * T35,'N       =',F12.7,T65,'D/D(N)       =',E15.6,//,
     * T35,'M       =',F12.7,T65,'D/D(M)       =',E15.6)
      WRITE(6,511) (I,X(I+5),I,G(I+5),I=1,NC)
  511 FORMAT(1H0,T35,'LOG(C',I1,')=',F12.7,T65,'D/D(LOG(C',I1,'))=',
     * E15.6)
      WRITE(6,512) V
  512 FORMAT(/,1H0,'FINAL STANDARD DEVIATION  :',F15.7)
C
      WRITE(6,513) (L(I),I=1,20)
  513 FORMAT(1H0,T5,'( COMMENT : ',20A4,' )')
C
      DO 50 J=1,NC
      JJ=J+5
      V=0.
      TD=0.
      IM=ND(J)+1
      IN=ND(J+1)
      K=ND(J+1)-ND(J)
      RK=FLOAT(K)
C
      DO 51 I=IM,IN
      W(I)=X(JJ)+X(4)*PLOG(I)+X(5)*TLOG(I)+ULOG(I)
      R=10.**(X(1)+X(2)*W(I)+X(3)*W(I)**2)
      IF(R.LE.70) GO TO 52
      TC(I)=0.
      GO TO 53
   52 CONTINUE
      TC(I)=EXP(-R)
   53 CONTINUE
      E(I)=TAU(I)-TC(I)
      PD(I)=100.*E(I)/TAU(I)
      TD=TD+ABS(E(I))
      V=V+E(I)**2
   51 CONTINUE
C
      TTD=TTD+TD
      TV=TV+V
      TD=TD/RK
      V=V/RK
      SF=SQRT(V)
      WRITE(6,514) (ID(J,I),I=1,9),K,(X(I),I=1,5),J,X(JJ),TD,V,SF
  514 FORMAT (1H1,T15,'ACTUAL/COMPUTED TRANSMITTANCES',////,
     * T10,'DATA   :',9A4,5X,'# OF POINTS =',I4,////,T10,
     * 'A1 =',F12.7,8X,'A2 =',F12.7,8X,'A3 =',F12.7,//,T10,'N  =',F12.7,
     * 8X,'M  =',F12.7,8X,'LOG(C',I1,') =',F12.7,////,
     * T15,'AVERAGE DISCREPANCY =',F12.7,//,
     * T15,'MEAN SQUARE ERROR   =',F12.7,//,
     * T15,'STANDARD DEVIATION  =',F12.7,////,
     * T5,'#',T11,'U =',T26,'P =',T36,'T =',T46,'X =',T56,'ACTUAL',
     * T65,'COMPUTED',T76,'DIFFERENCE',T90,'% DIFF.',/)
      WRITE(6,515) (I,U(I),P(I),T(I),W(I),TAU(I),TC(I),E(I),PD(I),
     *              I=IM,IN)
  515 FORMAT (1H ,I5,T10,E11.4,T24,F8.2,T34,F8.2,T43,F9.4,T53,F9.4,T63,
```

```
     * F9.4,T74,F11.6,T87,F10.4)
   50 CONTINUE
C
      FK=FLOAT(ND(NC+1))
      TTD=TTD/FK
      TV=TV/FK
      TSD=SQRT(TV)
      WRITE(6,516) ND(NC+1),TTD,TV,TSD
  516 FORMAT (1H1,///,T10,'TOTAL # OF POINTS USED       =',I6,///,T10,
     * 'GLOBAL AVERAGE DISCREPANCY    =',F12.7,///,T10,
     * 'GLOBAL MEAN SQUARE ERROR      =',F12.7,///,T10,
     * 'GLOBAL STANDARD DEVIATION     =',F12.7)
      RETURN
      END
      SUBROUTINE NMBC(X,NAME,NC,WN,CF,P0,T0)
C
C        COMPUTATION OF THE C'-VALUES FOR NON-MAJOR BANDS
C
      DIMENSION X(9),NAME(20),WN(4)
      DIMENSION CS(10),FS(10)
      DF=1.E30
      SGN=1.
      IF(X(3).LT.0.) SGN=-1.
C
C        IF THE QUADRATIC TERM IS TOO SMALL, THEN IT WILL BE IGNORED
C
      SMI=-2.*X(3)/X(2)
      IF(ABS(SMI).LE.1.E-6) GO TO 50
      SYM=1./SMI
   50 CONTINUE
C
      WRITE(6,5)(NAME(I),I=1,20)
    5 FORMAT(1H1,T15,20A4)
      WRITE(6,10)
   10 FORMAT(1H0,T15,' *** CALCULATION OF THE SPECTRAL PARAMETER FOR',
     * ' NON-MAJOR BANDS ***',///)
C
   11 CONTINUE
      NFREQ=0
   12 CONTINUE
      C=0.
      I=0
   15 CONTINUE
      READ(5,20,END=40) KGAS,FREQ,P,T,UGAS,TX
   20 FORMAT(I2,F10.3,E11.4,F9.3,24X,E11.4,F7.4)
      IF(KGAS.EQ.0) GO TO 25
      IF(KGAS.LT.0) GO TO 35
      IF(UGAS.GE.DF) GO TO 15
C
      I=I+1
      WX=FREQ
      UGAS=UGAS/CF
      IF(SMI.LE.1.E-6) GO TO 51
C
C        CASE 1  QUADRATIC TERM IS LARGE AND USED
C
```

```
      XS=SYM+SGN*ABS(SQRT(X(2)**2-4.*X(3)*(X(1)-ALOG10(-ALOG(TX))))
     *   /(2.*X(3)))
      GO TO 52
   51 CONTINUE
C
C        CASE 2  QUADRATIC TERM IS SMALL AND IGNORED
C
      XS=(ALOG10(-ALOG(TX))-X(1))/X(2)
   52 CONTINUE
      XC=X(4)*ALOG10(P/P0)+X(5)*ALOG10(T0/T)+ALOG10(UGAS)
      C=C+(XS-XC)
      GO TO 15
C
   25 C=C/FLOAT(I)
      NFREQ=NFREQ+1
      CS(NFREQ)=C
      FS(NFREQ)=WX
      DO 27 M=1,NC
      IF(ABS(WX-WN(M)).LE.0.1) CS(NFREQ)=X(5+M)
   27 CONTINUE
      IF(NFREQ.EQ.10) GO TO 30
      GO TO 12
   30 CONTINUE
      WRITE(6,31)(FS(K),K=1,NFREQ)
   31 FORMAT(1H0,2X,'WAVE NUMBER',2X,10F11.0)
      WRITE(6,32)(CS(K),K=1,NFREQ)
   32 FORMAT(1H0,5X,'C VALUES',2X,10F11.3//)
      GO TO 11
C
   35 CONTINUE
      IF(NFREQ.EQ.0) GO TO 40
      WRITE(6,31)(FS(K),K=1,NFREQ)
      WRITE(6,32)(CS(K),K=1,NFREQ)
   40 CONTINUE
      RETURN
      END
```

ED

80